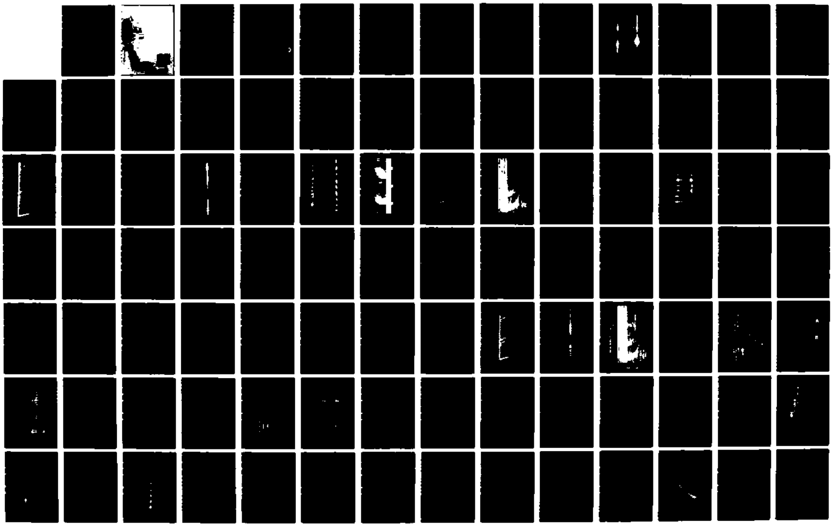
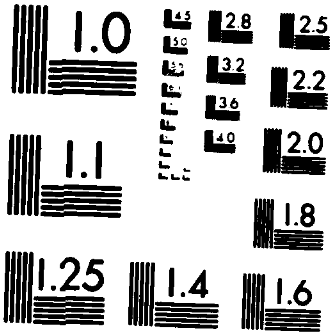
AD-A147 492 ARCTIC MARINE ACOUSTICS(U) LAMONT-DOHERTY GEOLOGICAL 1/3
OBSERVATORY PALISADES NY H W KUTSCHALE OCT 84
LDGO05-20081 N00014-80-C-0021

UNCLASSIFIED F/G 20/1 NL





MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS-1963-A

2

LAMONT-DOHERTY GEOLOGICAL OBSERVATORY OF COLUMBIA UNIVERSITY

Palisades, New York

10964

ARCTIC MARINE ACOUSTICS

by

Henry W. Kutschale

FINAL REPORT under Contract N00014-80-C-0021

Research Supported by:

Office of Naval Research
Arlington, Virginia 22217

October 1984

Abstract

Wave-theoretical computer codes have been developed to model pulse propagation in the central Arctic Ocean. Pulse shapes as a function of range and depth are computed from the Pulse Fast Field Program (PFFP) and the pulse parabolic equation (PPE) code. Group- and phase-velocity dispersion and eigenfunctions are computed from the PFFP or from a corresponding normal-mode program. Good agreement has been obtained between measured and computed SOFAR signals. The effect of ice roughness on Arctic SOFAR propagation is illustrated from field data and the PFFP. Hydroacoustic signals from underwater explosions that have propagated over the Arctic abyssal plains commonly display marked frequency dispersion in pulses that are bottom-interacting and that arrive after the SOFAR signal. In the infrasonic band of 2 to 20 Hz, the temporal dispersion for each pulse that has interacted with the flat bottom of the plain can be nearly as strong as that observed in the SOFAR signal for the first mode. However, the bottom-interacting pulses correspond to a coherent summation of many higher-order normal modes in a channel bounded above by the ocean surface and below by the zone of increasing velocity in the upper 400 m of the bottom sediment. Using normal-mode theory and the Multiple Scattering Pulse Fast Field Program (MSPFFP), we have analyzed the dispersion and pulse shapes and have derived the acoustic properties of the bottom in the Pole, Barents, and Mendeleyev Abyssal Plains. The principal properties of the bottom controlling the propagation are compressional velocity, density, and attentuation. In contrast, the ice layer has a negligible effect on the dispersion of the observed waves. The effect on pulse compression of this frequency dispersion of the bottom-interacting signals was simulated numerically, using predistorted waveforms matched to the dispersion of the SOFAR channel at specified ranges. We include a users manual (with test computations) for our computer codes. We hope that this manual will be useful to the many users of our codes.

Introduction

Our research in Arctic Marine Acoustics has been a strong interplay between field observations and computer modeling of underwater propagation. In the spring of 1982 at FRAM IV/TRISTEN we completed the fourth consecutive field season of acoustic measurements in the Eurasian Basin. Project TRISTEN for methods of long-range signaling in the Arctic channel was spawned by our research in long-range propagation. The new experiments of the FRAM series were highly successful, and they provided us with the first opportunity to obtain systematic data over two abyssal plains of infrasonic signals (2 to 20 Hz) traveling via bottom-interacting paths to long-ranges in the Arctic channel. The frequency dispersion of these signals was of practical interest in connection with TRISTEN, but our main objective was to understand the propagation of these waves in the Arctic channel and to derive the acoustic properties of the upper 400 m of sediment which supports the propagation of these waves to long ranges.

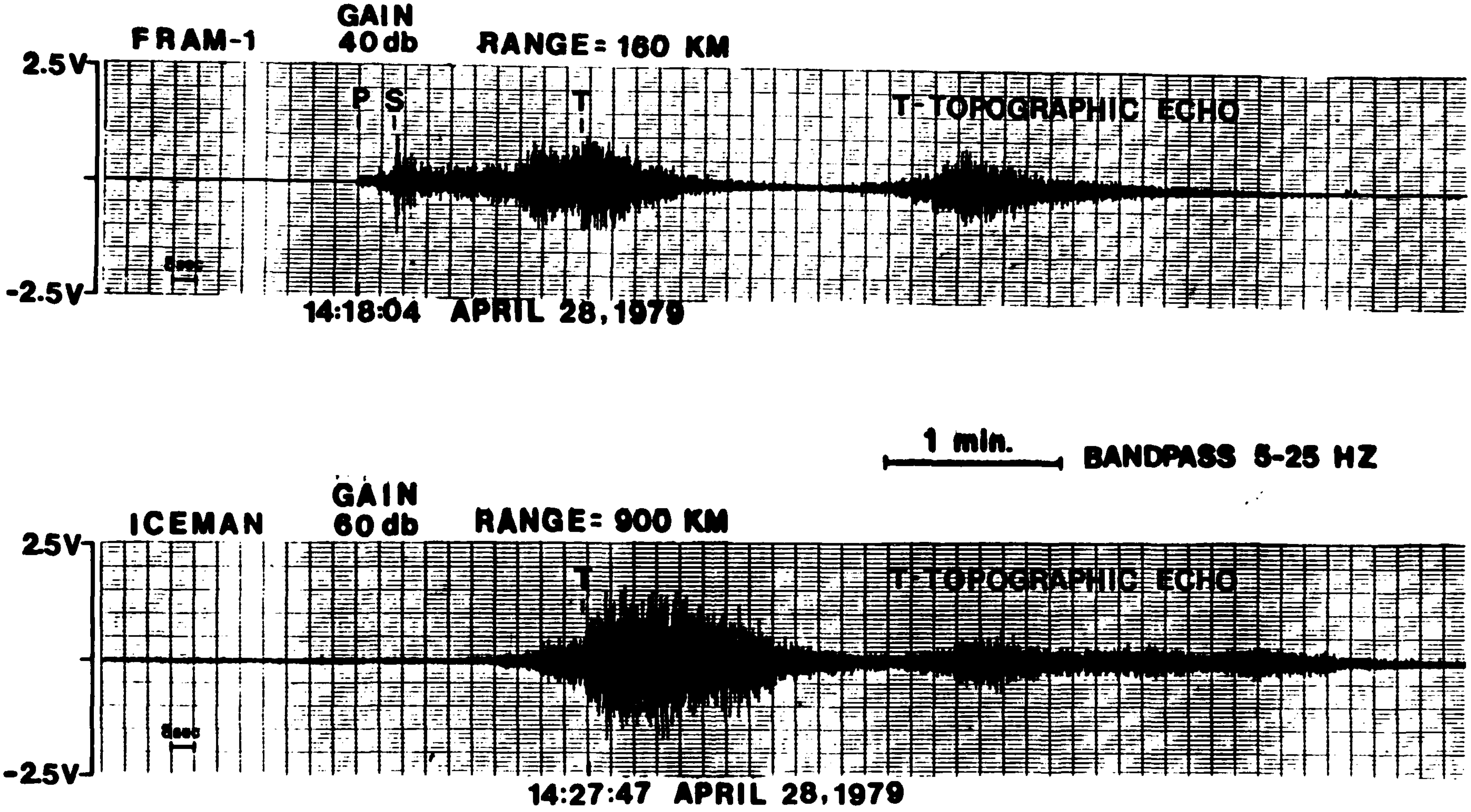
The new experiments also provided us with a vastly expanded data set for a systematic investigation of environmental factors controlling the dispersion and excitation of the normal modes observed in Arctic SOFAR propagation. We also had, for the first time, the opportunity to investigate long-range propagation of hydoracoustic signals from earthquakes at the Nansen Ridge (Fig. 1).

Simultaneously with the field measurements and data analysis, we were improving and standardizing our full-wave computer models for transmission loss and pulse propagation to make them as automatic and fool-proof as possible so that we could conveniently analyze the wave propagation. We have completed this task with the interim standard codes, and we have distributed the most recent updates of the combined FFP and normal-mode program (and the corresponding pulse code) to the Navy users group.

The Arctic SOFAR signals commonly differ markedly in charcter from those observed in the deep channel, largely because of the predominance of very low frequency waves in the Arctic channel. Waves above 100 Hz are strongly scattered by the rough ice boundaries, and thus they are not propagated within the waveguide to great ranges. Because the wavelengths of the observed waves are commonly comparable with the dimensions of the portion of the duct controlling the propagation, full wave theory must be used to explain the propagation, particularly for waves traveling in the upper thousand meters of the water column and for those waves with turning points in the upper 400 m of bottom sediment. The observations of marked frequency dispersion of explosion sounds at long ranges motivated the development of the present computer codes.

In this report we summarize our research in two invited papers presented at the Acoustial Society of America. We include a users manual (with test computations) for our computer codes. We hope that this manual will be useful to the many users of these codes within the Naval community.

Fig. 1 Typical earthquake signal recorded at FRAM I and LOREX Camp Iceman. P, S, and T phases shown in figure.



Bibliography

Kutschale, H.W., B. Allen and C. Monjo, "Acoustic Wave Propagation," Arctic Geophysics and Oceanography: LOREX and FRAM I, EOS Trans. Am. Geophys. Union, 61(17) 0 246, 279, 1980. (Abstract)

Keller, J.B., D.M. Levy and D.S. Ahluwalia, "Internal and Surface Wave Production in a Stratified Fluid," Wave Motion, 3, North-Holland Publishing Company, 215-229, 1981.

Kutschale, H.W., "Pulse propagation in the Central Arctic Ocean," J. Acoust. Soc. Am., Supp. 1, 69, p. S19, 1981. Invited paper, manuscript available from author.

Kutschale, H.W. and T. Lee, "Bottom-interacting Acoustic Signals in the Arctic Channel: Long-Range Propagation," J. Acoust. Soc. Am., Supp. 1, 74, p. S1, 1983. Invited paper, manuscript available from authors.

Lee, T., "Long-range Sound Transmission of Bottom-Interacting Signals in the Arctic Channel: Experiment and Theory," Ph.D. Dissertation, Graduate School of Arts and Sciences, Columbia University, New York, NY 10027, 1983.

PULSE PROPAGATION OF THE CENTRAL ARCTIC OCEAN

Henry W. Kutschale

Lamont Doherty Geological Observatory
of Columbia University
Palisades, New York 10964

Invited Paper

Presented at the 101st Meeting

Acoustical Society of America

Ottawa, Ontario, Canada

May 18 - 22, 1981

Abstract

Wave-theoretical computer codes have been developed to model pulse propagation in the central Arctic Ocean. Pulse shapes as a function of range and depth are computed from the Pulse Fast Field Program (PFFP) and the pulse parabolic equation (PPE) code. Group- and phase-velocity dispersion and eigenfunctions are computed from the PFFP or from a corresponding normal-mode program. Good agreement has been obtained between measured and computed SOFAR signals. Computer simulations of pulse compression show in detail the possibility of achieving significant signal gain by pulse compression in the Arctic channel. This is accomplished by using predistorted waveforms matched to the dispersion of the channel at specified ranges. The effect of ice roughness on Arctic SOFAR propagation is illustrated from field data and the PFFP. Recent experiments of long-range transmission of explosion sound over the Pole Abyssal Plain show marked frequency dispersion of bottom-interacting signals following the SOFAR signal. The dispersion of each pulse interacting with the flat ocean bottom of the Plain at a depth of 4200 m can be nearly as strong as that observed in the SOFAR signal for the first mode, but the bottom-interacting pulses appear to correspond to very high-order normal modes. Present efforts are described to model these bottom-interacting pulses, as well as T-phase propagation from earthquakes at the Nansen Ridge.

PULSE PROPAGATION IN THE CENTRAL ARCTIC OCEAN

by

Henry W. Kutschale

Lamont-Doherty Geological Observatory
of Columbia University
Palisades, N. Y. 10964

Introduction

The two features of the polar environment that most strongly influence underwater sound are the permanent ice cover and the velocity structure in the water. The ice modifies propagation, particularly at high frequencies, by scattering waves from the rough ice boundaries. Sound velocity is increasing with depth from the surface to the bottom. In sufficiently deep water, sounds are transmitted to great ranges in this sound channel by upward refraction in the water and repeated reflection from the ice canopy. The surface sound channel of the Arctic is the polar extension of the deep sound channel or SOFAR channel of the nonpolar oceans (Ewing and Worzel, 1948), but the Arctic signals are often quite different in character from those observed in the deep channel, largely because of the predominance of low-frequency waves (5 to 40 Hz) in the Arctic channel.

The experiments of SOFAR propagation in the central Arctic Ocean span a twenty-three year period. This work was initiated by NUSC in 1958-59. Lamont cooperated in the field with NUSC during these early experiments.

These early experiments revealed the unique features of Arctic SOFAR propagation. Along a 1000 km path, the explosion signals were confined to a frequency band from about 7 to 35 Hz. The signals showed marked frequency dispersion. The first two normal modes were commonly observed. These waves were successfully interpreted by normal-mode theory for the Arctic channel (Kutschale, 1961).

Frequency dispersion of explosion sounds in shallow water was discovered by Ewing (1936) and was extensively investigated by Worzel and Ewing (1948) during World War II. These observations of Worzel and Ewing in shallow water were explained by Pekeris (1948) who had developed the normal-mode theory of shallow-water propagation. There is a close analogy between propagation in shallow water and propagation to long ranges in deep water of the central Arctic Ocean. In the central Arctic Ocean, the dispersion is different from that observed in shallow water because the Arctic propagation is controlled by the velocity structure in the water, but the methods of analyzing the Arctic signals parallels closely the methods employed by these early workers.

The Waveguide

The central Arctic Ocean is covered by ice the year round. The pack ice averages about 3 meters thick. The ice is in continual, very slow motion under the influence of winds and currents. The pack ice is not of uniform thickness, but it is broken by leads of open water and pressure ridges. The irregularity of the pack-ice boundaries is dependent upon location and time of year, but a pattern of ice roughness useful to underwater sound has been found for broad areas of the Arctic Ocean for each of the four seasons of the year (Diachok, 1973).

This seasonal and spatial dependence of ice roughness is of great importance for long-range sound propagation because of scattering of waves impinging upon the ice. We illustrate this effect in Figures 1 and 2. In Figure 1 we show a computer simulation of pulse shapes as a function of range with no ice roughness in the model. In Figure 2 we do the same simulation but with an RMS ice roughness of 20 m. The amplitude scale is the same as in the previous Figure 1, but we note the marked decrease in the amplitudes of the waves, particularly the high-frequency components of waves corresponding to the first normal mode. We have never observed such rough ice in the Arctic, but this extreme example was chosen to illustrate a dramatic effect on the waves at moderately short ranges.

In contrast to the variability of the ice cover, the sound-speed structure with depth in the water is very uniform as a function of location and season. Sound speed is an increasing function of depth from the surface to the bottom. Sounds are transmitted to long ranges in this sound channel by upward refraction in the water and repeated reflection from the ice. Figure 3 shows a typical Arctic profile. The waveguide consists basically of three parts: the upper layer, the lower layer, and the bottom interface. The strong gradient to a depth of 300 m is controlled by the rapid increase of temperature to this depth, while the deeper, weaker gradient is controlled by the pressure effect. The profile is approximately bilinear with the sharp decrease in the gradient near 350 m in the Amerasian Basin and 300 m in the Eurasian Basin (Crary, 1956).

Figure 4 shows typical ray paths computed at one-degree intervals from a source near the axis of the Arctic channel. The deep penetrating RSR waves arrive first, and the waves traveling near the axis last. From this diagram we see that the onsets of the signals are variable with range in amplitude and mean horizontal velocity in response to the cycling of the deep penetrating RSR sounds, but that waves traveling near the surface terminate the pulse abruptly. This is shown in Figure 5 by a detailed computer simulation of pulse shapes as a function of range. Topographic features will block the deep-penetrating RSR waves. The highest frequency waves correspond to the deep-penetrating RSR paths. These waves make fewer contacts with the ice per unit horizontal distance than the shallower paths. The bottom reflections commonly correspond only to the lowest frequency waves observed in the signal because of attenuation of waves at both boundaries of the waveguide. The principal complications for making detail comparisons between experiment and theory are introduced by wave scattering from the ice and the bottom.

If the bottom is irregular along the propagation path, we generally neglect the bottom interactions in the computer simulations. This assumption is justified from field data such as shown in Figure 6. For this series of experiments, shots of identical size and depth were launched from T-3 and recorded aboard ARLIS II employing a hydrophone at a fixed depth. Note the weakening of the deep-penetrating waves as the amount of shallow water along the path increases. This experiment shows qualitatively that the onset of a signal is largely determined by waves which pass over all bottom topography without suffering bottom reflections. Following the abrupt termination of the SOFAR signal, we see weak waves scattered from the bottom. Propagation of these incoherent scattered waves is beyond the capabilities of our present computer codes which model only the coherent portion of the pulses.

At low frequencies theories for the coherent component of the scattered waves from rough surfaces account adequately for attenuation of waves reflected from the ice. Both the formulation by Kuperman (1975) and the one by Mellen and Marsh (1965) are used in the computer codes.

Data and Some Comparisons with Theory

Figures 7 and 8 show typical Arctic SOFAR signals at a range of 1300 km and 720 km from the shot. Three wave groups are shown clearly by Figure 8. The first group corresponds to the deep-penetrating RSR waves. This group lasts about four seconds on this oscillogram. The first two pulses of this group are well separated in time, but toward the end of this group, at four seconds on the oscillogram, it is difficult to resolve the individual arrivals. Following this group of waves we observe a strong train of nearly sinusoidal waves in Figures 7 and 8 showing clear frequency dispersion, with low-frequency waves traveling faster than high-frequency waves in this wavetrain. These waves correspond to the first normal mode propagated near the axis of the channel. Following the abrupt termination of the SOFAR signal, we observe weak bottom reflections rising above an oscilliatory wavetrain corresponding to waves scattered from an irregular bottom.

If the bottom is flat over the whole propagation path, we observe many strong coherent bottom-interacting pulses at long ranges. These pulses following the SOFAR signal can show masked frequency dispersion. This is shown in Figure 9 from a typical signal propagated along a 316 km path over the Pole Abyssal Plain which is at a depth of 4200 m below the surface. The oscillogram and the spectrogram are not quite on the same time scale, and thus we have indicated corresponding arrivals on the two displays of the signal. The upper trace of the oscillogram corresponds to a slightly lower amplifier gain than the lower trace. In this Figure 9 we clearly see the frequency dispersion of the first two normal modes of the SOFAR signal. Following the abrupt termination of the SOFAR signal, we observe sixteen strong bottom-interacting pulses. The total duration of the SOFAR signal and these bottom-interacting pulses is nearly fifty seconds. The bottom-interacting pulses near the end of the signal are nearly sinusoidal.

In Figure 10 we show in detail the waveforms and spectra of two of the bottom interacting pulses of the preceding Figure 9. The signals are normally dispersed, with low-frequency waves traveling faster than high-frequency waves. We are presently trying to explain the dispersion and pulse shapes of these waves from a model which extends the vertical velocity profile of the ocean several hundred meters into the bottom of the Plain. We believe that each bottom-interacting pulse corresponds to portions of dispersion curves for very high-order normal modes derived from such a model. This conclusion is based upon our many years of experience in analyzing dispersion of the normal-modes of the SOFAR signals.

The sound spectrograms of Figures 11 and 12 show the frequency dispersion of two typical SOFAR signals. We see the dispersion clearly of waves corresponding to the first and second normal modes. Waves of the second normal mode were difficult to see on the preceding oscillograms because the amplitudes are considerably weaker than waves of the first normal mode. The sharp pulses on Figure 8 corresponding to the deep-penetrating RSR waves show horizontal bands across the spectrogram of Figure 12, but a close examination of many spectrograms shows that these pulses correspond to a superposition of many higher-order normal modes with group velocities equal to the mean horizontal velocities of each of the pulses. We should point out that these spectrograms give only a qualitative measure of the relative amplitudes of the waves. Our primany purpose for this type of display is to resolve the normal modes of the signal.

In Figure 13 observed dispersion is compared with computed dispersion for the first two modes. The data were taken from Figure 12. We have converted the group-velocity scale of the computations to a time scale to make a direct comparison with the observed spectrogram possible. The measured and computed dispersion curves virtually coincide for both modes. For these computations we have used the sound-speed profile of Figure 3 which was measured during the transmission experiment in the Eurasian Basin.

Figure 14 shows oscillograms from three shots which illustrate the effect of source depth on waves of the first mode. We note that waves corresponding to the first normal mode are weak for the 366-meter shot. This is due to the pressure distribution with depth of the first normal mode. Each oscillogram corresponds to one output from a hydrophone and one from a vertical-component geophone on the ice surface.

Figure 15 shows reasonably good agreement between theory and experiment of the depth variation of pressure for the first normal mode at the two frequencies of 20 and 40 Hz.

In Figure 16 we show the effect of ice roughness on the peak amplitudes of the waves of the first normal mode as a function of range. This experiment was conducted in the Amerasian Basin over the Canada Abyssal Plain. We analyzed the waveforms in the band from 25 to 70 Hz because the effect of ice roughness on propagation is far stronger in this band than at lower frequencies. The shots were dropped from a Navy aircraft on a straight path to the listening station. The sound sources were 1.8 lb SUS charges fired at 800 feet. When we compare the peak intensity as a function of range for waves of the first normal mode, we get good agreement between the measurements and the computer simulations of the pulses if we use an RMS underice roughness of 3 m. It was gratifying to learn subsequent to these experiments that about the same value of RMS roughness had been derived in this area from direct measurements of ice roughness.

Pulse Compression

The uniformity of dispersion data of waves corresponding to the first normal mode in the frequency band 8 to 30 Hz suggests the possibility of achieving significant signal gain by pulse compression in the Arctic channel (DiNapoli, Viccione, and Kutschale, 1978). This is shown from computer simulations by using predistorted waveforms matched to the dispersion of the channel at specified ranges.

The pulse compression technique was originally developed for radar systems to obtain high range resolution with long pulses. In that application, the frequency of the transmitted pulse is varied with time to produce a wideband waveform. A matched filter is then implemented at the receiver by passing the waveform through a filter with the inverse frequency versus time relationship. All frequency components of the pulse are thus properly delayed and appear simultaneously at the output to produce a pulse of shorter duration and high peak amplitude (Walter, 1961).

The technique was first used in underwater acoustics by Parvalescu and Clay (1965) to examine signal transmission stability in Tongue of the Ocean. Their experiments were performed at 400 Hz in 1.8 km of water at a range of 36 km. They discovered that the signal enhancement was realizable but sensitive to geometry (source and receiver depth variations). The Arctic offers a far better channel (one strong mode at 15 Hz versus over a hundred modes at 400 Hz) for the application of this scheme.

Figure 17 shows computed SOFAR signals in the range interval from 300 to 500 km. The source and hydrophone are at a depth of 100 m. Near the beginning of the signals the second mode is strongest, followed by the strong waves of the first normal mode. Note the clear frequency dispersion in the signals which was shown in Figure 13 to be in close agreement with field data. The peak amplitude of the waves decreases by spherical spreading since the effect of ice roughness is small at these frequencies to ranges of 500 km. We have assumed in these simulations that the propagation path was over an irregular bottom so that the bottom-interacting signals shown previously from the Pole Abyssal Plain could be neglected.

The computer codes are highly automated, and we can thus investigate pulse compression as a function of range by reversing in time pulses such as those shown in Figure 17 and propagating them back through the channel. Figure 18 shows the evolution of pulse compression as a function of range. The input signal at the source is the time reverse of the 400 km signal of Figure 17. The ocean filter is matched at 400 km. This is shown by the sharp pulse at this range. We note that the peak amplitude of the 400 km signal is greater than the peak amplitude of the 300 km signal. Thus the desired effect has been achieved. At 500 km the signal rapidly decreases in amplitude as the pulse spreads apart. At 300 km the dispersion in the main pulse is opposite to that characteristic of the channel, but at 500 km the pulse has started to spread with the normal dispersion of the channel. The sidelobes of the compressed pulse correspond to crosscorrelations between the modes. Note the frequency dispersion in the sidelobes. This dispersion reverses on either side of the central pulse.

If we had included all the bottom interactions in the simulations, we would expect to observe many more sidelobes corresponding to cross-correlations between the normal modes of the water column and those affected by the bottom. The effect of bottom-interacting signals on pulse compression is currently under investigation.

The degradation with range of the time reversed waveform is graphically depicted in Figure 19 by plotting the peak amplitude of the received waveform versus range. With no special signal shaping, the peak amplitude would decay with range according to 20 log R. This is due to the combined effects of cyclindrical spreading and also a stretching of the waveform in time. The initial decay of the peak amplitude with range for the time-reversed signal is seen to obey a 10 log R dependence. This occurs because as the range increases the pulse is trying to compress itself and one is left with only cyclindrical spreading. Beyond the matched range, the pulse again begins to spread in time and the peak amplitude obeys the 20 log R decay law.

Acknowledgment

This work was supported by the Office of Naval Research under Contract N00014-80-C-0021 with the Lamont-Doherty Geological Observatory of Columbia University.

Appendix A

Computer Codes

Pulse shapes as a function of range and depth are computed from the Pulse Normal Mode Program (PNMP) (Kutschale, 1969 and 1970), the Pulse Fast Field Program (PFFP) (Kutschale and DiNapoli, 1977), and the Pulse Parabolic Equation (PPE) code (Kutschale and Tappert, 1977). Group - and phase - velocity dispersion and eigenfunctions are computed from the PFFP or from the corresponding normal - mode program. Each of these pulse codes uses Fourier synthesis via the fast Fourier transform to coherently add up the frequency response at each frequency point of the broad-band input signal to compute the temporal waveforms as a function of range.

I will not go into the mathematical details of each of the models. I have given a list of references which provide some of the details. One might ask why have we developed three computer codes. The answer is that each code enjoys certain advantages over the others depending upon the application.

For example, the PPE will accommodate a range dependent environment, and we can investigate the effects on the waveforms of small variations in the observed sound-speed profiles as well as model a range-dependent ice roughness. However, for most applications we can neglect lateral variations in the sound-speed profiles and the ice roughness.

The PFFP and the PNMP are each based upon the same integral representation of the sound field in a horizontally stratified ocean. Either of these codes can accommodate a stratified medium of a mixture of solid layers and liquid layers. Thus we can investigate the effects of the ice sheet on the propagation, or we can calculate waveforms measured by geophones on or in the ice. Furthermore, these codes are appropriate for investigating the bottom-interacting signals since no paraxial approximation is made to the full wave equation. In the PFFP the integral representation for each frequency component is integrated directly over wavenumber, while in the PNMP the integral representation for each frequency component of the pulse is expanded in a normal-mode series plus a branch-line integral. The contribution of the branch-line integral is negligible at long ranges. If we obsdrve only a few modes and if we want waveforms at only a few range points, the PNMP is very fast compared to the PPE or PFFP, which effectively computes all modes over the specified range of phase velocities and frequencies.

In the PNMP or PPE the scattering effect of the ice on the amplitudes of the waves in the pulses is modeled by using a modified formula of Mellen and Marsh (1965). In the PFFP we use either this approach or, since damping in each layer is a natural feature of this code, we use a very convenient formulation of scattering by Kuperman (1975).

In Figures 1A and 2A we show by comparison of waveforms the effects on the signals of neglecting some of the modes. For the PFFP simulation we have included the full contribution to the signal of waves with phase velocities between the surface sound velocity and 1490 m/sec in the sound-speed profile of Figure 3. In the PNMP simulation we have computed the signal corresponding to only the first four modes. We see from this comparison that the ends of the wavetrains are identical but that the beginnings differ in detail because in order to model a sharp pulse, corresponding to the deep-penetrating RSR waves shown at a range of 50 km in Figure 1A and on the measured signal of Figures 8 and 12, we must include all the higher modes. When this is done, both the PFFP and PNMP give identical results.

REFERENCES

Crary, A. P. Arctic ice island research. Advances in Geophysics, 3, Academic Press, New York, 1-39, 1956.

Diachok, O. I. Preliminary empirical model of low frequency under-ice reflection loss. NAVOCEANO Technical Note No. 6130-3-73, 1973.

DiNapoli, F. R. Fast field program for multilayered media. Naval Underwater Systems Center, Report No. 4103, 1971.

DiNapoli, F. R., D. Viccione and H. Kutschale. The Arctic channel: an acoustic waveguide. Naval Underwater Systems Center, New London, CT., Technical Memorandum No. 781078, 1978.

DiNapoli, F. R. and R. L. Deavenport. Numerical Models of underwater acoustic propagation. Topics in Current Fhysics, Vol. 8: Ocean Acoustics, J. A. DeSanto (ed.), Springer-Verlag, New York, 1979.

Ewing, M. Frequency of water waves. U. S. Coast and Geodet. Survey Field Engrs. Bull., p. 65, 1936.

Ewing, M. and J. L. Worzel. Long-range sound transmission. Propagation of Sound in the Ocean, Geol. Soc. Am., Mem. 27, 1-35, 1948.

Kuperman, W. A. Coherent component of specular reflection and transmission at a randomly rough two-fluid interface. J. Acoust. Soc. Am., 58, 365-370, 1975.

Kutschale, H. W. Long-range sound transmission in the Arctic Ocean. Journal of Geophysical Research, 66:2189-98, 1961.

Kutschale, H. W. Arctic hydroacoustics. ARCTIC, 22:246-264, 1969.

Kutschale, H. W. The integral solution of the sound field in a multilayered liquid-solid half space with numerical computations for low-frequency propagation in the Arctic Ocean. Lamont-Doherty Geol. Obs. of Columbia University, Technical Report No. 1, 1970.

Kutschale, H. W. Rapid computation by wave theory of propagation loss in the Arctic Ocean. Lamont-Doherty Geol. Obs. of Columbia University, Technical Report No. 8, 1973.

Kutschale, H. W. and F. D. Tappert. Pulse propagation in the ocean. I: The parabolic equation method. J. Acoust. Soc. Am., 62, Supplement No. 1, G5, 1977.

Kutschale, H. W. and F. R. DiNapoli. Pulse propagation in the ocean. II: The fast field program method. J. Acoust. Soc. Am., 62, Supplement No. 1, G6, 1977.

Mellen, R. H. and H. W. Marsh. Underwater sound in the Arctic Ocean. New London, CT: U. S. Navy Underwater Sound Laboratory, Report MED-65-1002, 1965.

Parvulescu, A. and C. S. Clay. Radio Elec. Eng., 29:223-228, 1965.

Pekeris, C. L. Theory of propagation of explosive sound in shallow water. Propagation of Sound in the Ocean, Geological Society of America Memoir, 27:1-117, 1948.

Pekeris, C. L. Propagation of seismic pulses in layered liquids and solids. International Symposium on Stress Wave Propagation in Materials, Norman Davids (ed.), Interscience Publishers, Inc., New York, 45-57, 1960.

Tappert, F. Selected applications of the parabolic-equation method in underwater acoustics. International Workshop on Low-Frequency Propagation and Noise, Vol. 1, 155-197, Maury Center for Ocean Science, Washington, D. C., 1977.

Tappert, Fred D. The parabolic approximation method. Wave Propagation and Underwater Acoustics, Joseph B. Keller and John S. Papadakis (ed.), 70, 224-287, 1977.

Walter, K. Pulse compression in an acoustic waveguide. J. Acoust. Soc. Am., 33, 681-686, 1961.

Worzel, J.L. and M. Ewing. Explosion sounds in shallow water. Propagation of Sound in the Ocean, Geological Society of America Memoir, 27:1-53, 1948.

Captions of Figures

Figure 1. Computed Arctic SOFAR signals. No ice roughness.

Figure 2. Computed Arctic SOFAR signals. RMS ice roughness 20 m.

Figure 3. Sound speed as a function of depth in Eurasian Basin of the Eastern Arctic Ocean.

Figure 4. Ray paths computed at a one-degree interval.

Figure 5. Computed Arctic SOFAR signals as a function of range.

Figure 6. Signals showing the effect of bathymetry.

Figure 7. Waveform of typical Arctic SOFAR signal measured in Amerasian Basin. Range 1300 km.

Figure 8. Waveform of typical signal measured in Eurasian Basin. Range 720 km.

Figure 9. Waveform and corresponding spectrogram of typical signal measured over flat bottom of the Pole Abyssal Plain. Water depth 4100 m. Range 316 km.

Figure 10. Waveforms and corresponding spectrogram of two bottom interacting signals of Figure 9.

Figure 11. Sound spectrogram of typical SOFAR signal measured in Amerasian Basin. Range 870 km.

Figure 12. Sound spectrogram of typical signal measured in Eurasian Basin. Range 720 km.

Figure 13. Comparison of measured and computed group-velocity dispersion for first two normal modes.

Figure 14. Waveforms as a function of depth. Range about 600 km.

Figure 15. Comparison of measured and computed vertical pressure dispersion for waves of the first normal mode.

Figure 16. Comparison of experiment and theory of peak intensity as a function of range of waves of the first normal mode. RMS ice roughness 3 m.

Figure 17. Computed SOFAR signals as a function of range.

Figure 18. Waveforms of pulse compression as a function of range.

Figure 19. Peak amplitude of compressed pulse as a function of range.

Figure 1A. Waveforms computed by PFFP corresponding to inclusion of all normal modes.

Figure 2A. Waveforms computed by PNMP. Only first four normal modes computed.

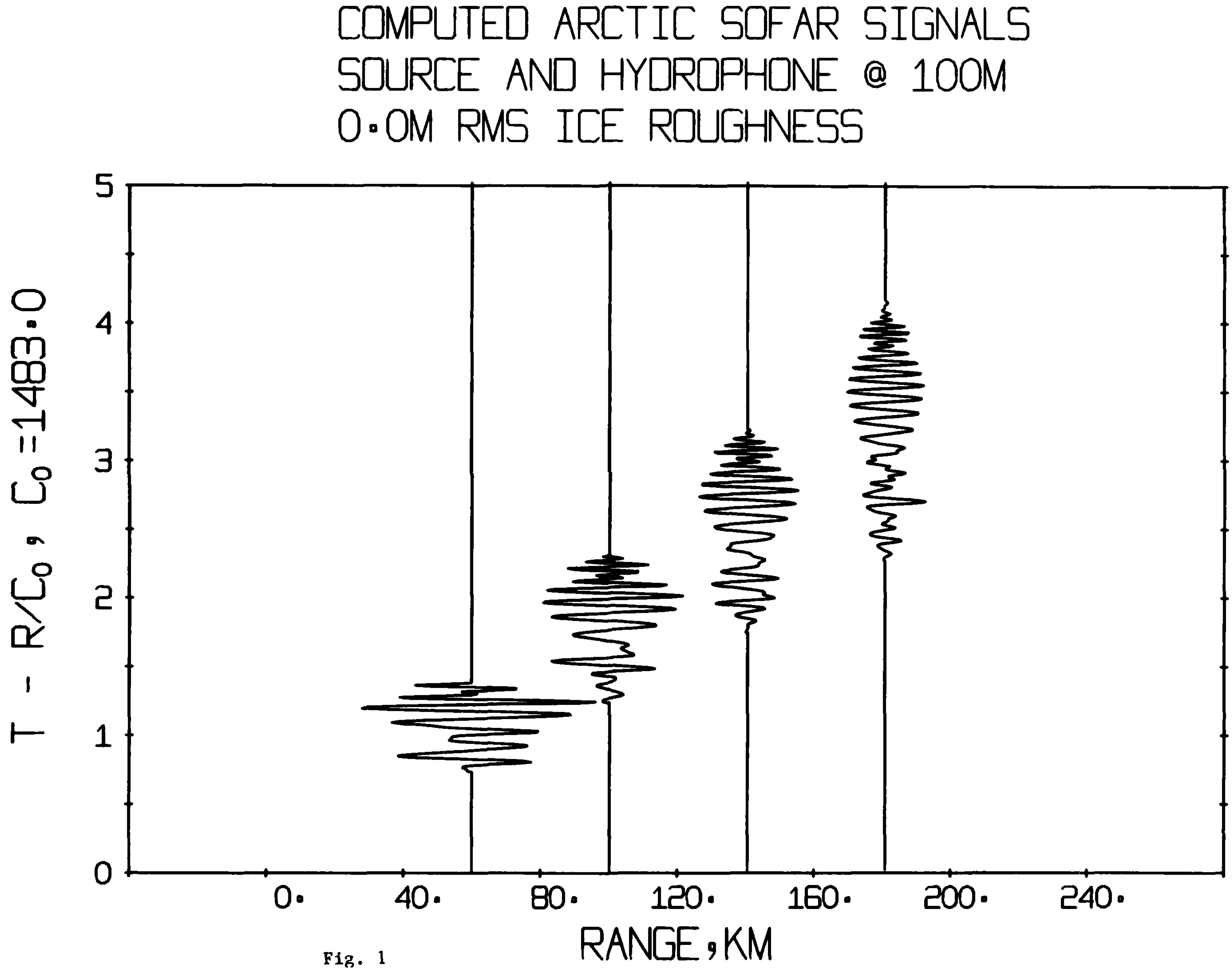

COMPUTED ARCTIC SOFAR SIGNALS
SOURCE AND HYDROPHONE @ 100M
0.0M RMS ICE ROUGHNESS
T - R/C$_0$, C$_0$ =1483.0
5
4
3
2
1
0
0.
40.
80.
120.
160.
200.
240.
RANGE,KM

Fig. 1

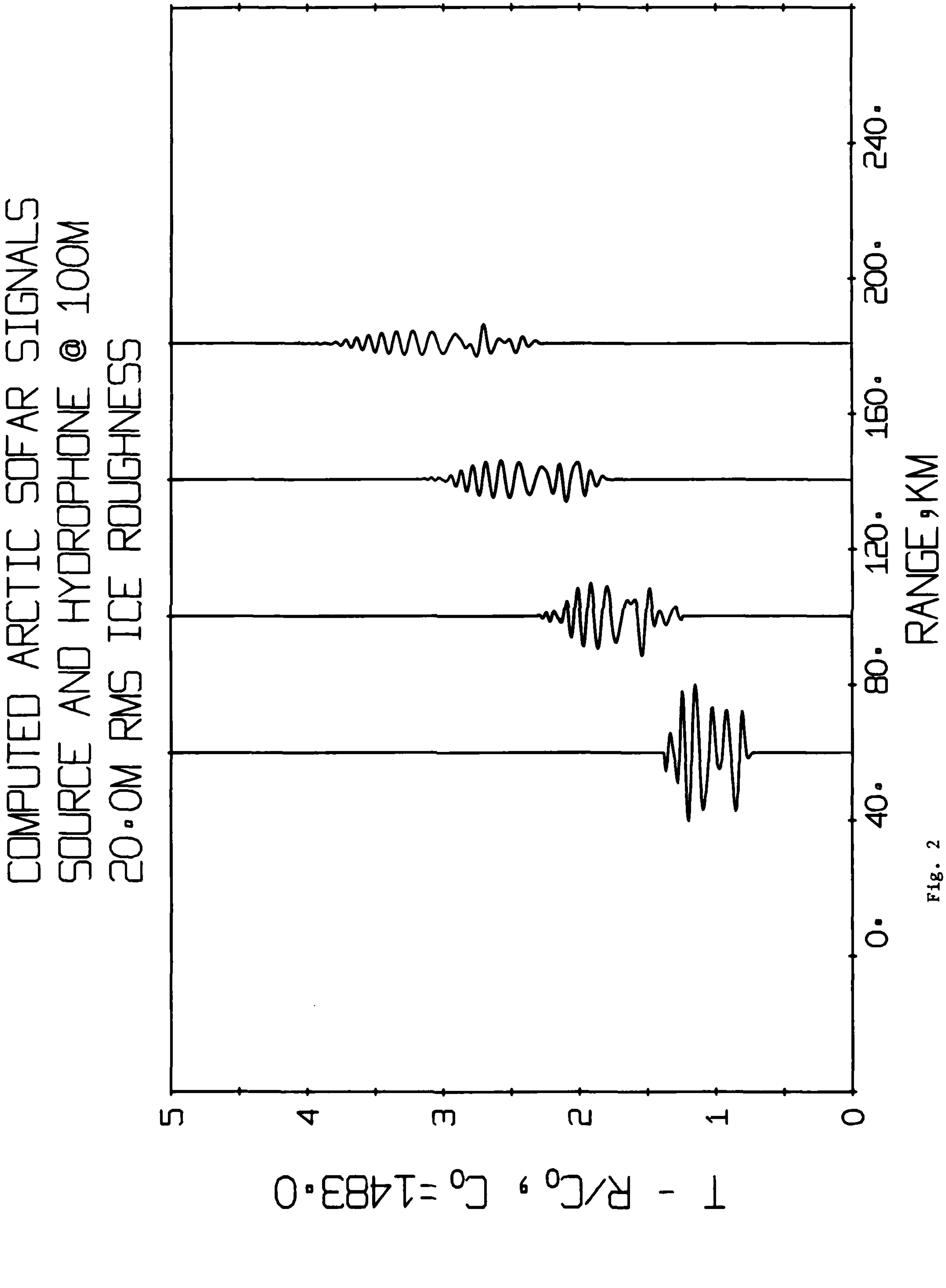

COMPUTED ARCTIC SOFAR SIGNALS
SOURCE AND HYDROPHONE @ 100M
20.0M RMS ICE ROUGHNESS
0.
40.
80.
120.
160.
200.
240.
RANGE, KM
0
1
2
3
4
5
T - R/C₀, C₀ =1483.0

Fig. 2

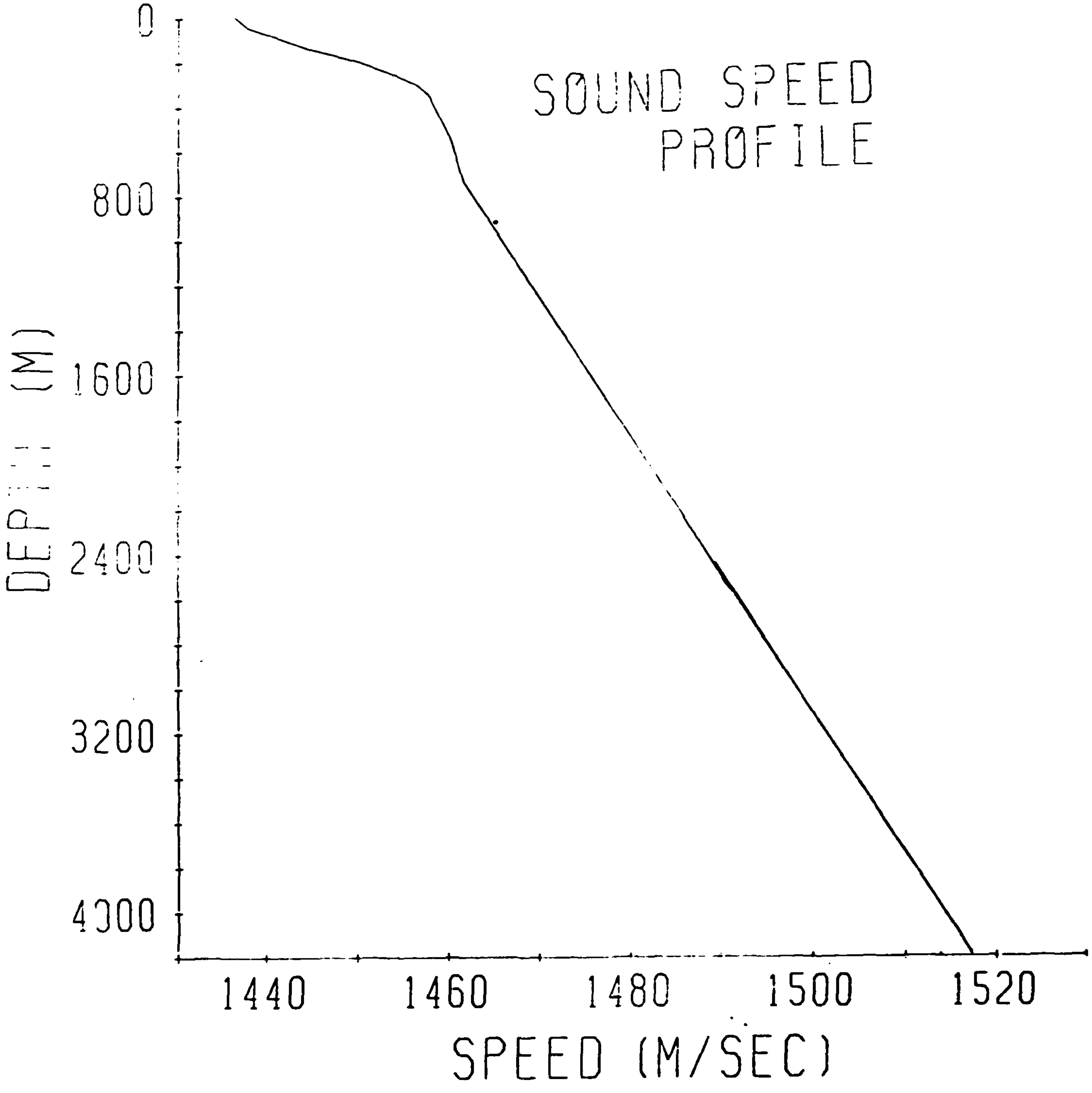
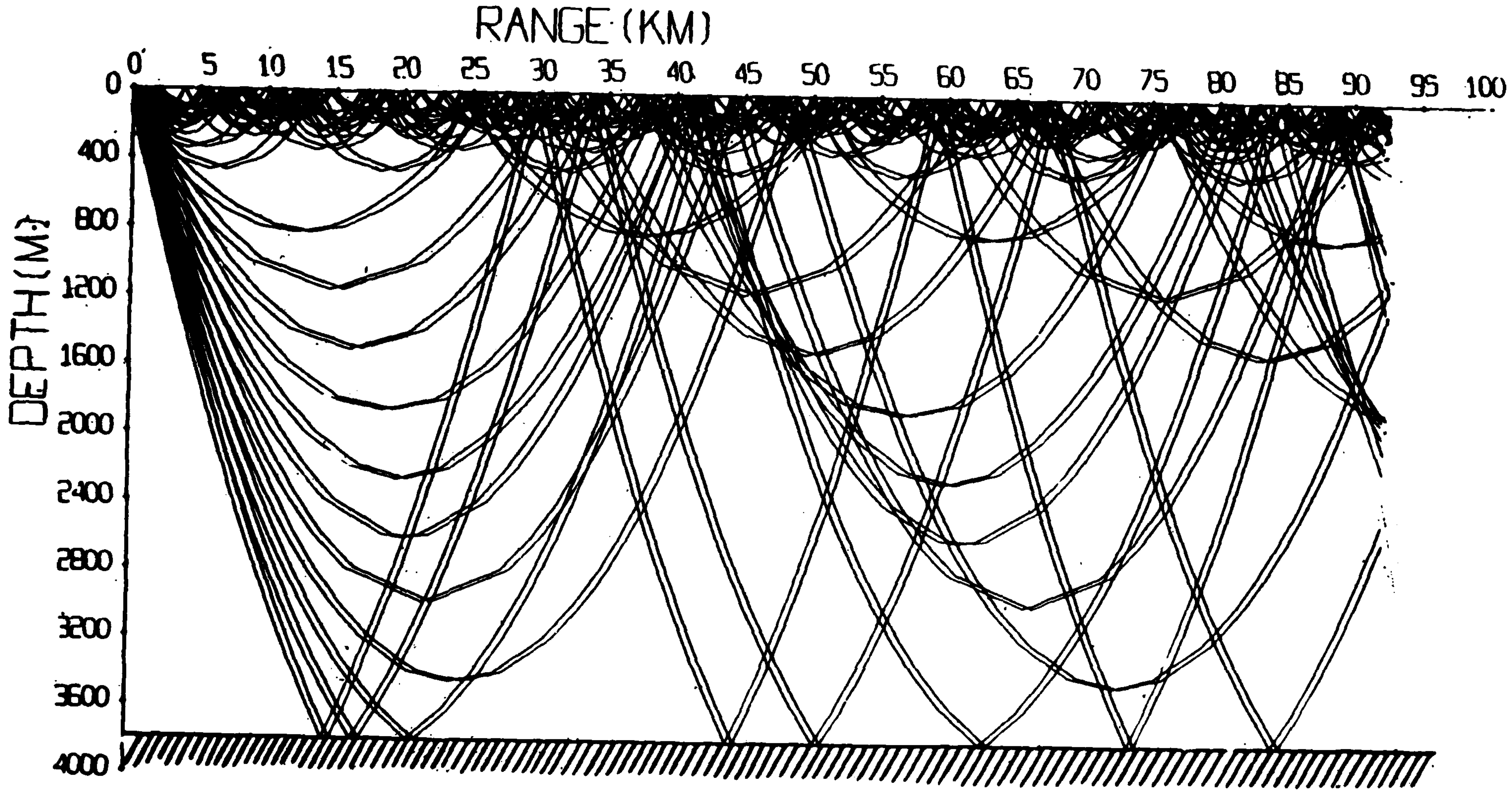

SOUND SPEED
PROFILE
0
800
1600
2400
3200
4000
DEPTH (M)
1440
1460
1480
1500
1520
SPEED (M/SEC)

Fig. 3

Fig. 4



RAY PATHS FOR ARCTIC CHANNEL, 1° INTERVAL

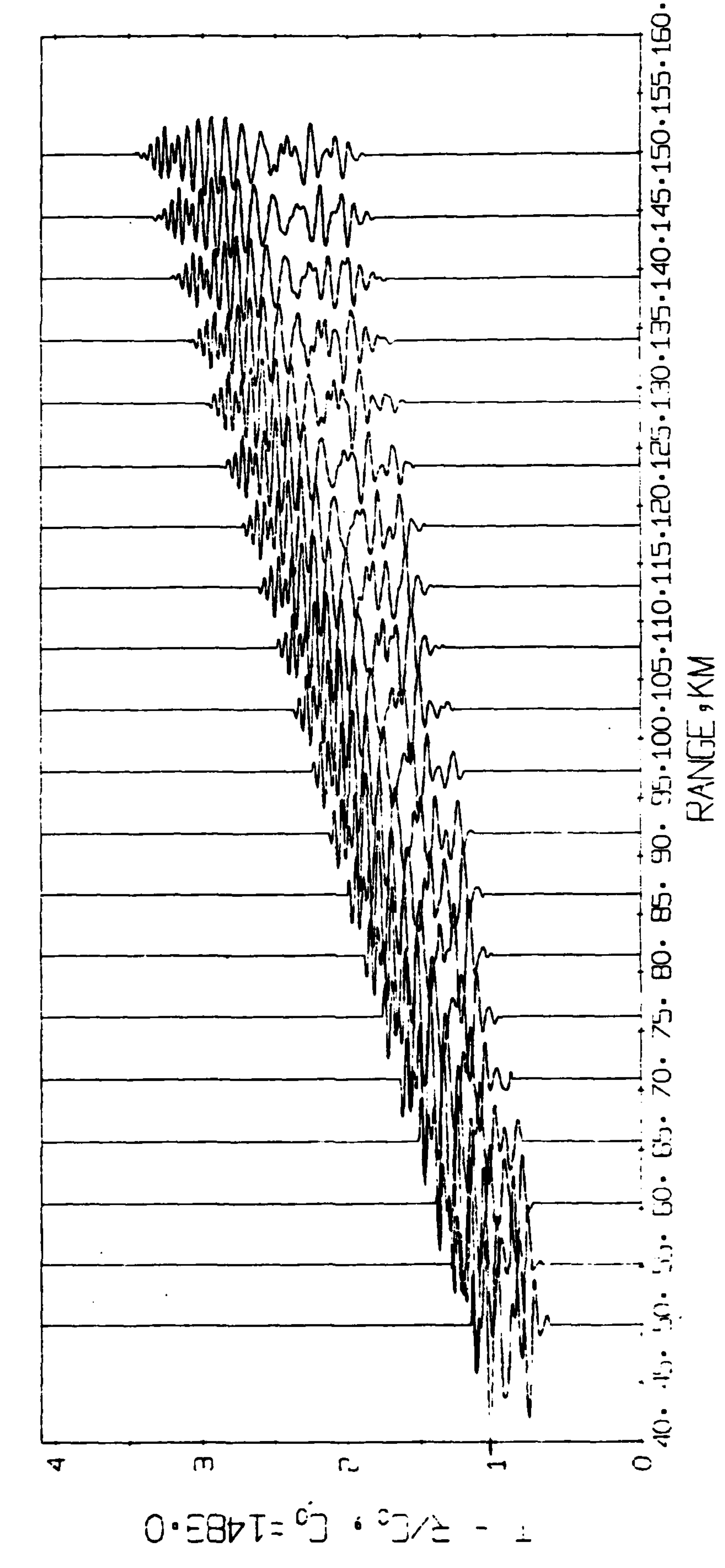

40· 45· 50· 55· 60· 65· 70· 75· 80· 85· 90· 95·100·105·110·115·120·125·130·135·140·145·150·155·160·
RANGE,KM
0
1
2
3
4
T - R/C0 , C0 =1483.0

Fig. 5

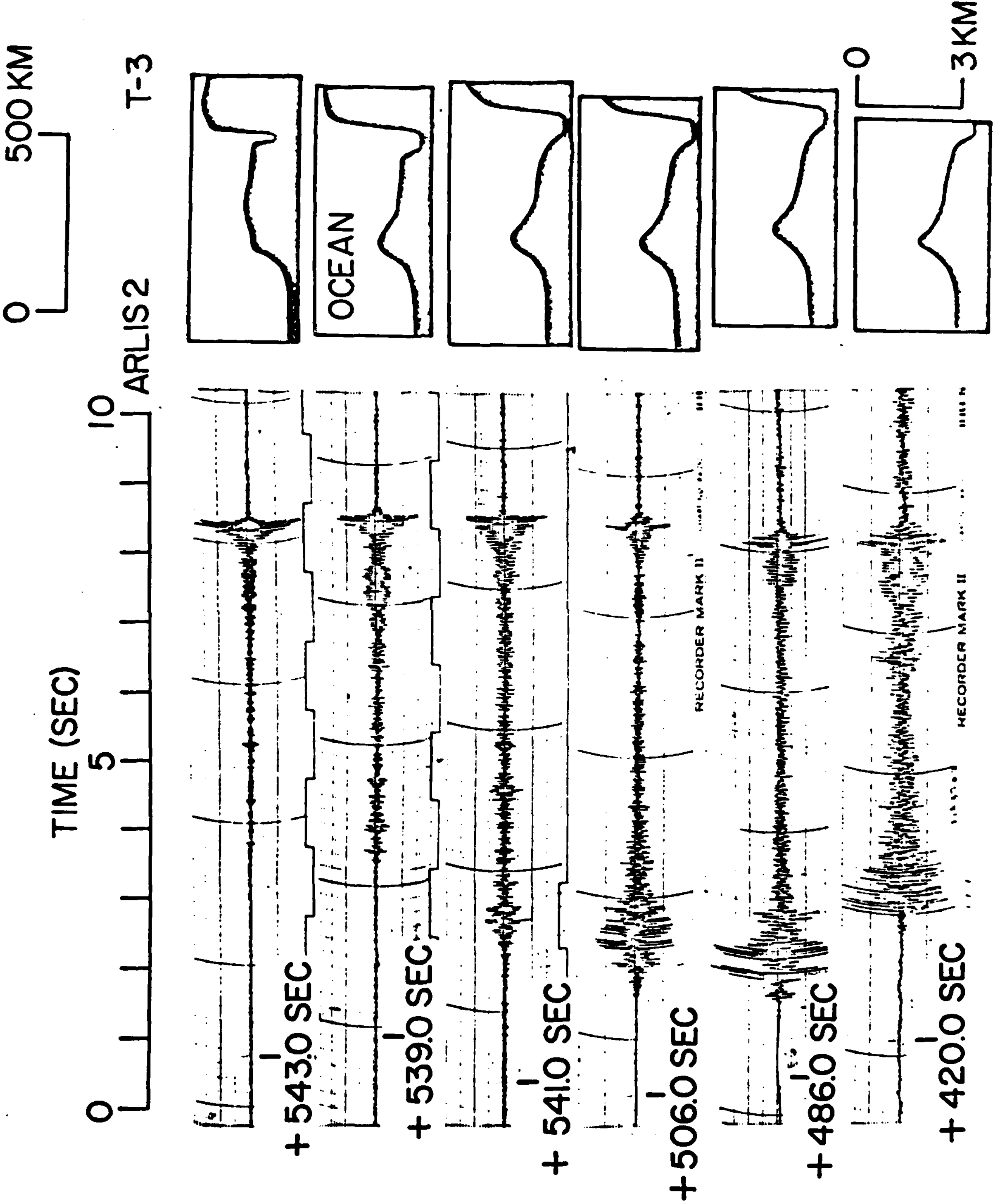
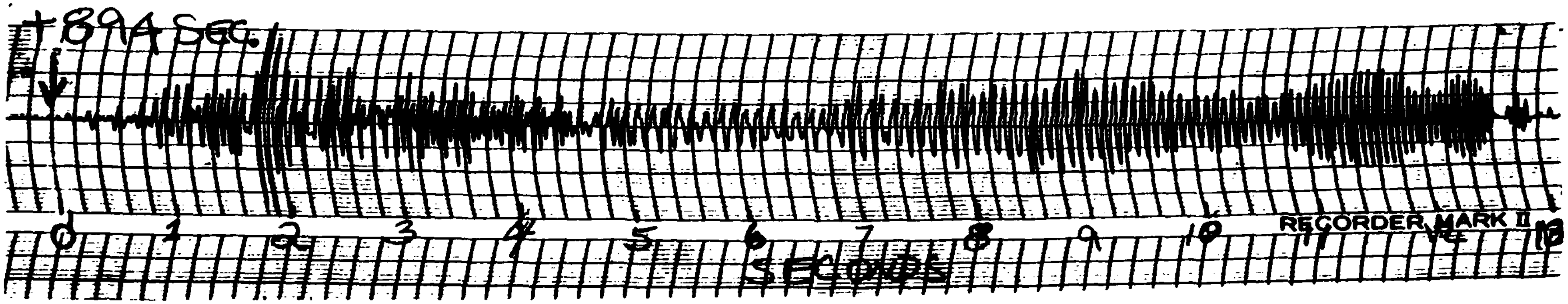

TIME (SEC)
0
5
10
0
500 KM
ARLIS 2
T-3
OCEAN
0
3 KM
+ 543.0 SEC
+ 539.0 SEC
+ 541.0 SEC
+ 506.0 SEC
+ 486.0 SEC
+ 420.0 SEC
RECORDER MARK II
RECORDER MARK II
RECORDER MARK II

Fig. 6

Fig. 7



ARCTIC SOFAR
SIGNAL. 1300 KM.

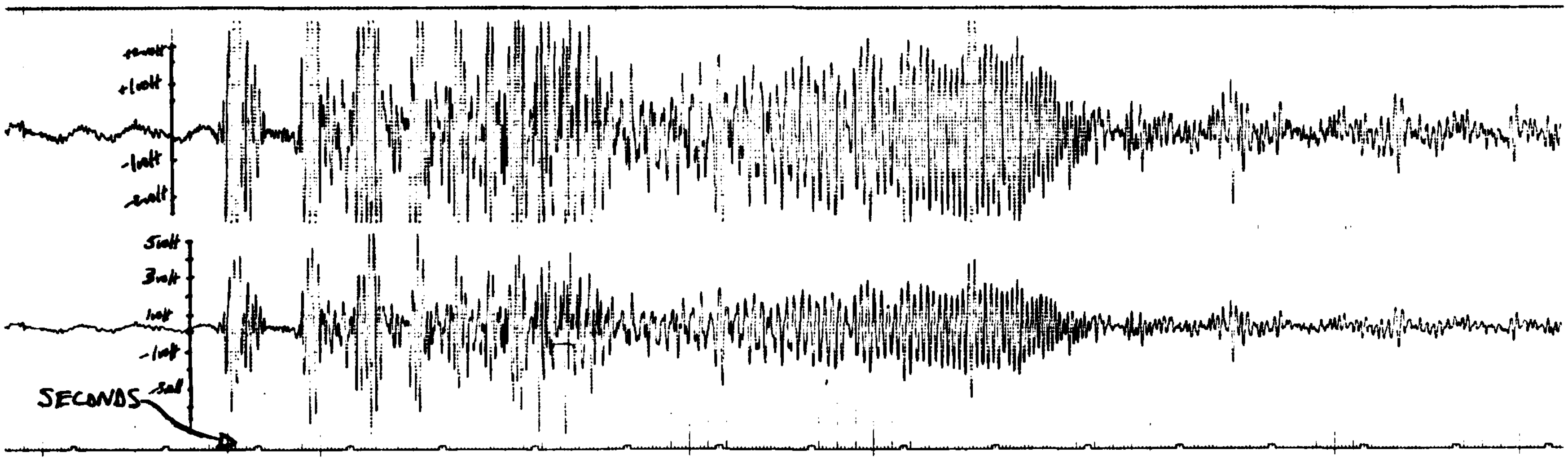
FRAM I ACOUSTIC DATA

TAPE # 10

800' 55 LB + ONE SUS charge

Hydrophone at 175'

TOP: 50db at 100mv/DIV



BOTTOM: 50db at 200mv/DIV

Fig. 8

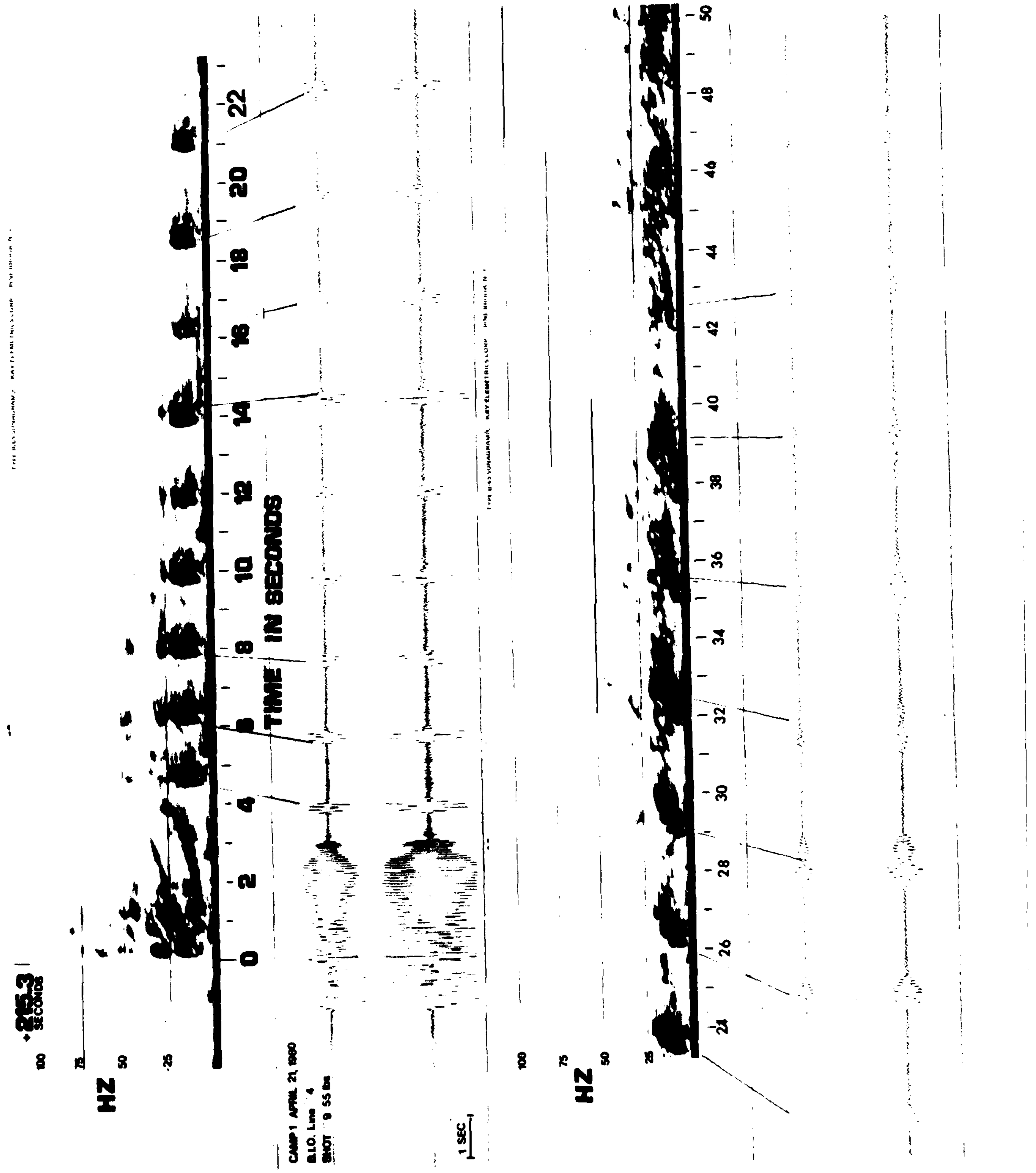

+215.3 SECONDS
HZ
100
75
50
25
0
2
4
6
8
10
12
14
16
18
20
22
TIME IN SECONDS
CAMP 1 APRIL 21, 1980
B.I.O. Line 4
SHOT 9 55 lbs
1 SEC
KAY ELEMETRICS CORP. PINE BROOK N.J.
HZ
100
75
50
25
24
26
28
30
32
34
36
38
40
42
44
46
48
50

Fig. 9

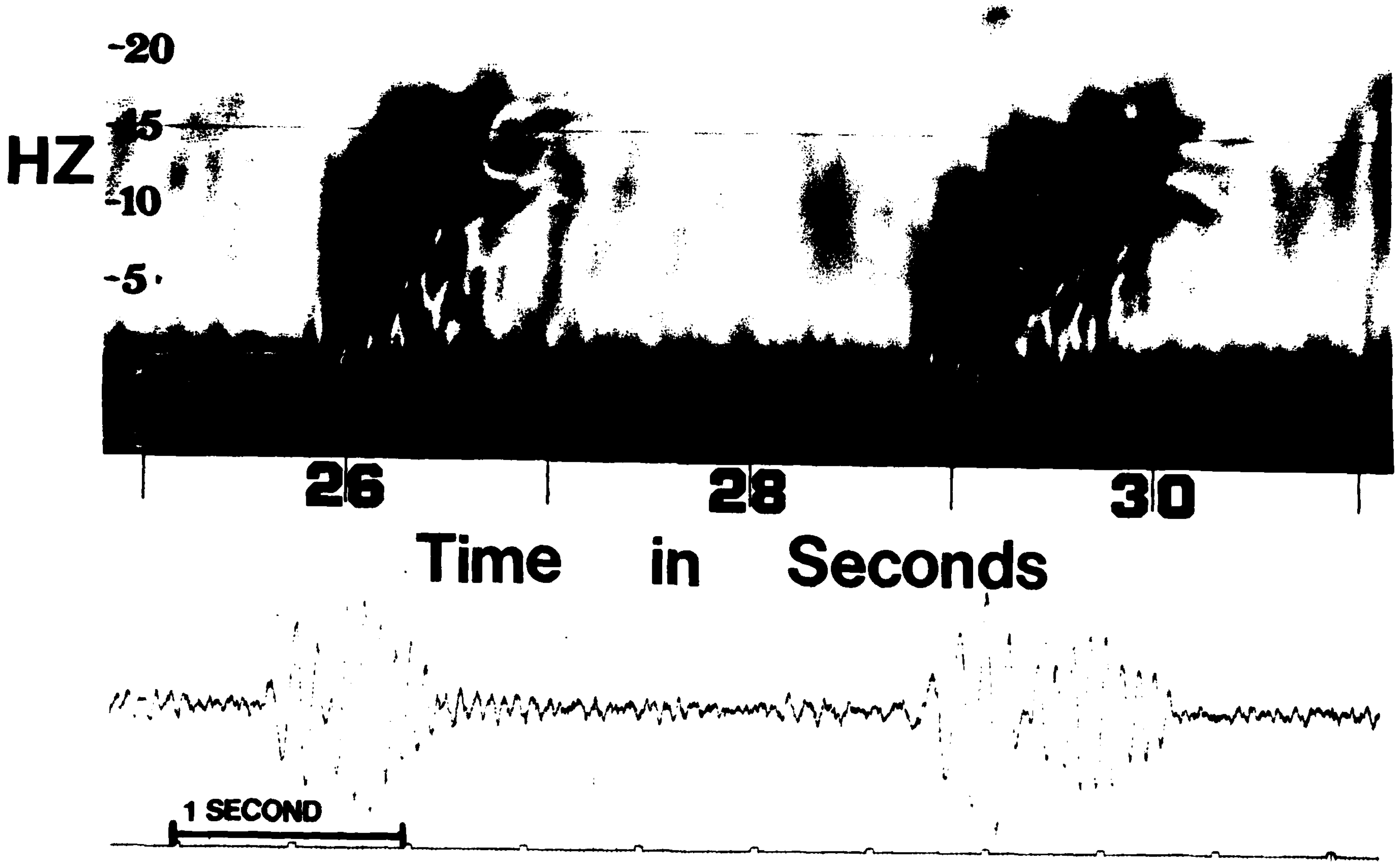


Fig. 10

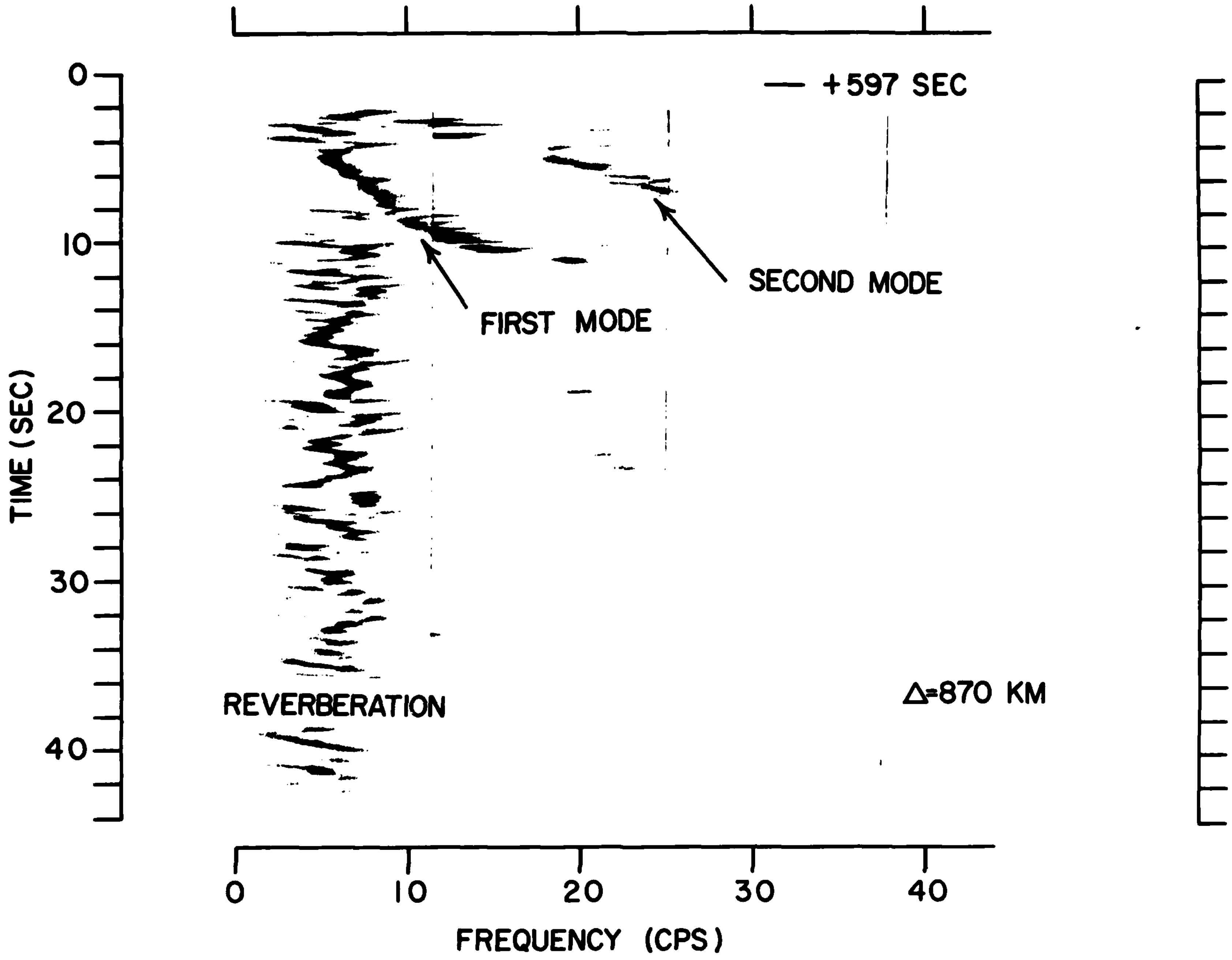

— +597 SEC
SECOND MODE
FIRST MODE
REVERBERATION
Δ=870 KM
TIME (SEC)
0
10
20
30
40
FREQUENCY (CPS)
0
10
20
30
40

Fig. 11

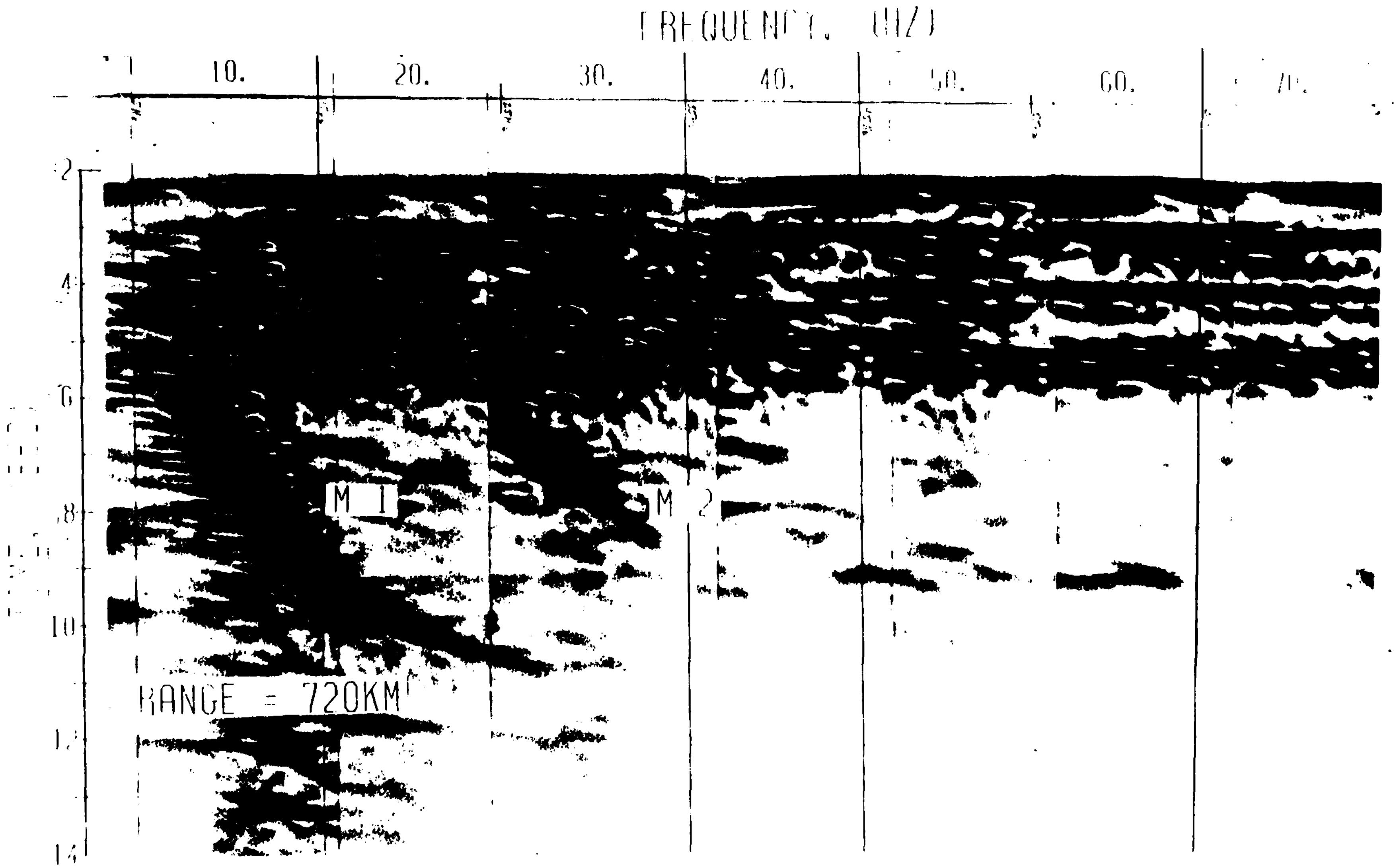


Fig. 12

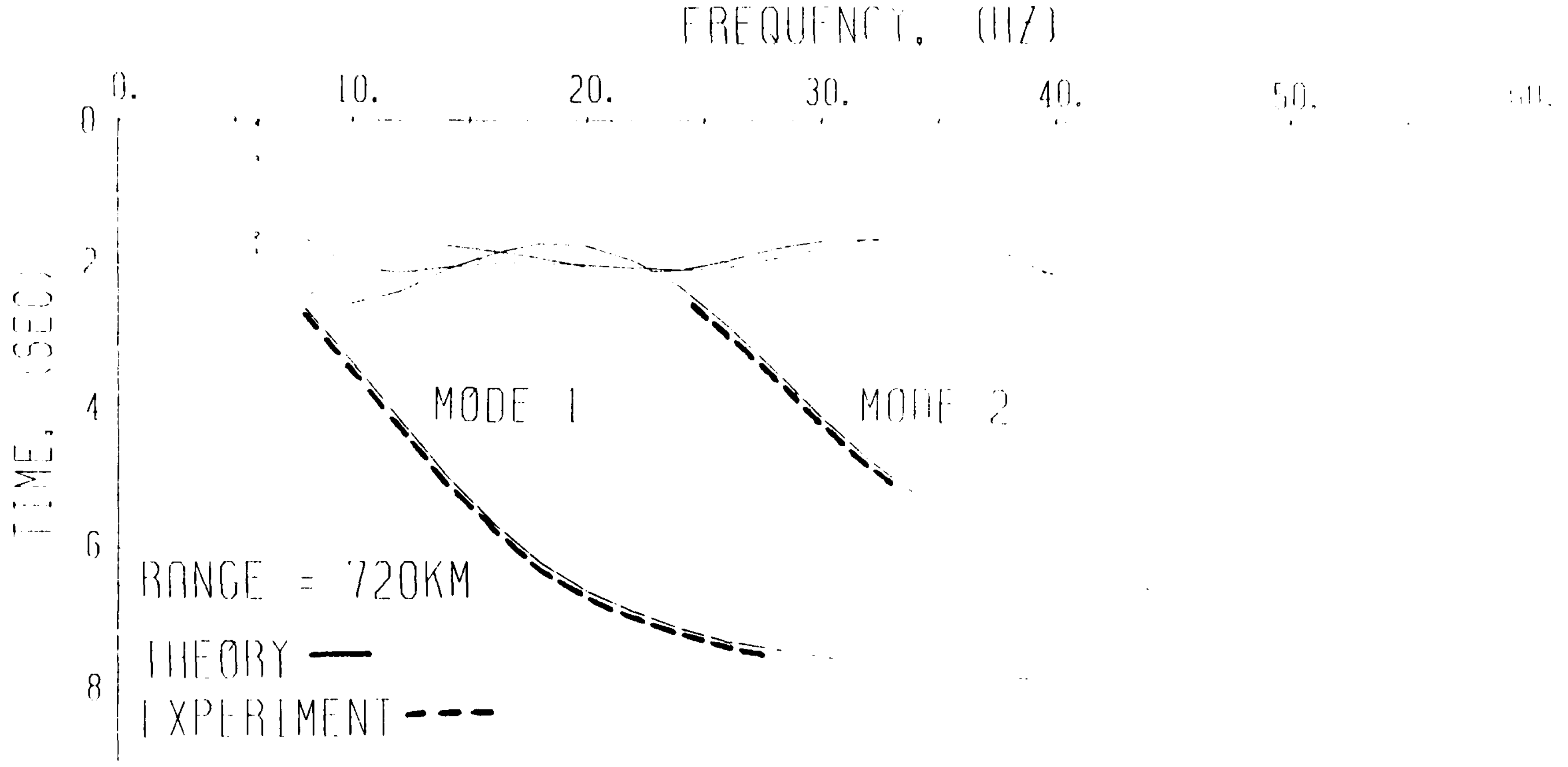

FREQUENCY, (HZ)
0.
10.
20.
30.
40.
50.
60.
TIME, (SEC)
0
2
4
6
8
MØDE 1
MODE 2
RANGE = 720KM
THEØRY —
EXPERIMENT - - -

Fig. 13

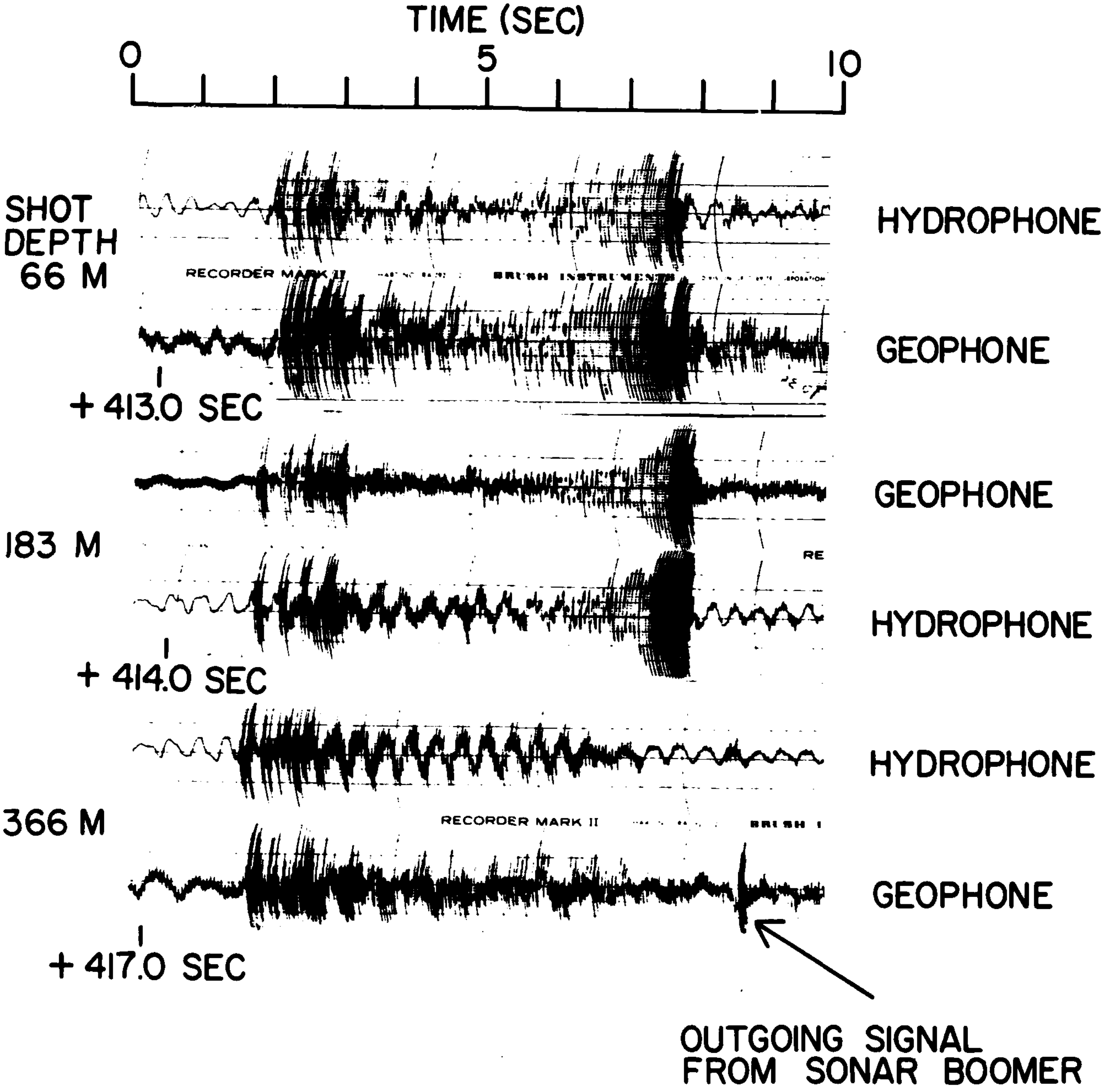

TIME (SEC)
0
5
10
SHOT DEPTH 66 M
HYDROPHONE
RECORDER MARK II
BRUSH INSTRUMENTS
GEOPHONE
+ 413.0 SEC
GEOPHONE
183 M
HYDROPHONE
+ 414.0 SEC
HYDROPHONE
366 M
RECORDER MARK II
GEOPHONE
+ 417.0 SEC
OUTGOING SIGNAL FROM SONAR BOOMER

Fig. 14

PRESSURE-DEPTH VARIATION

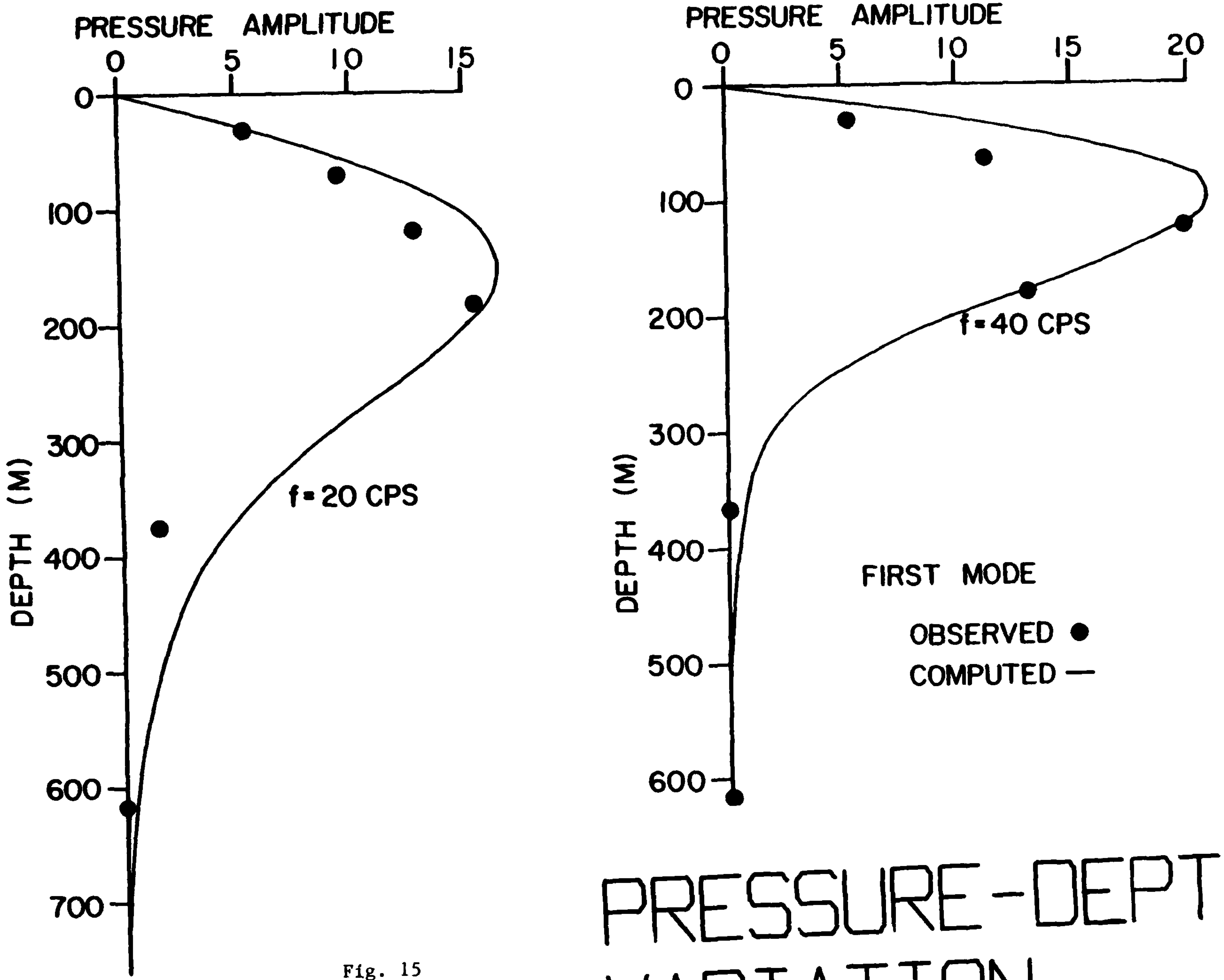


Fig. 15

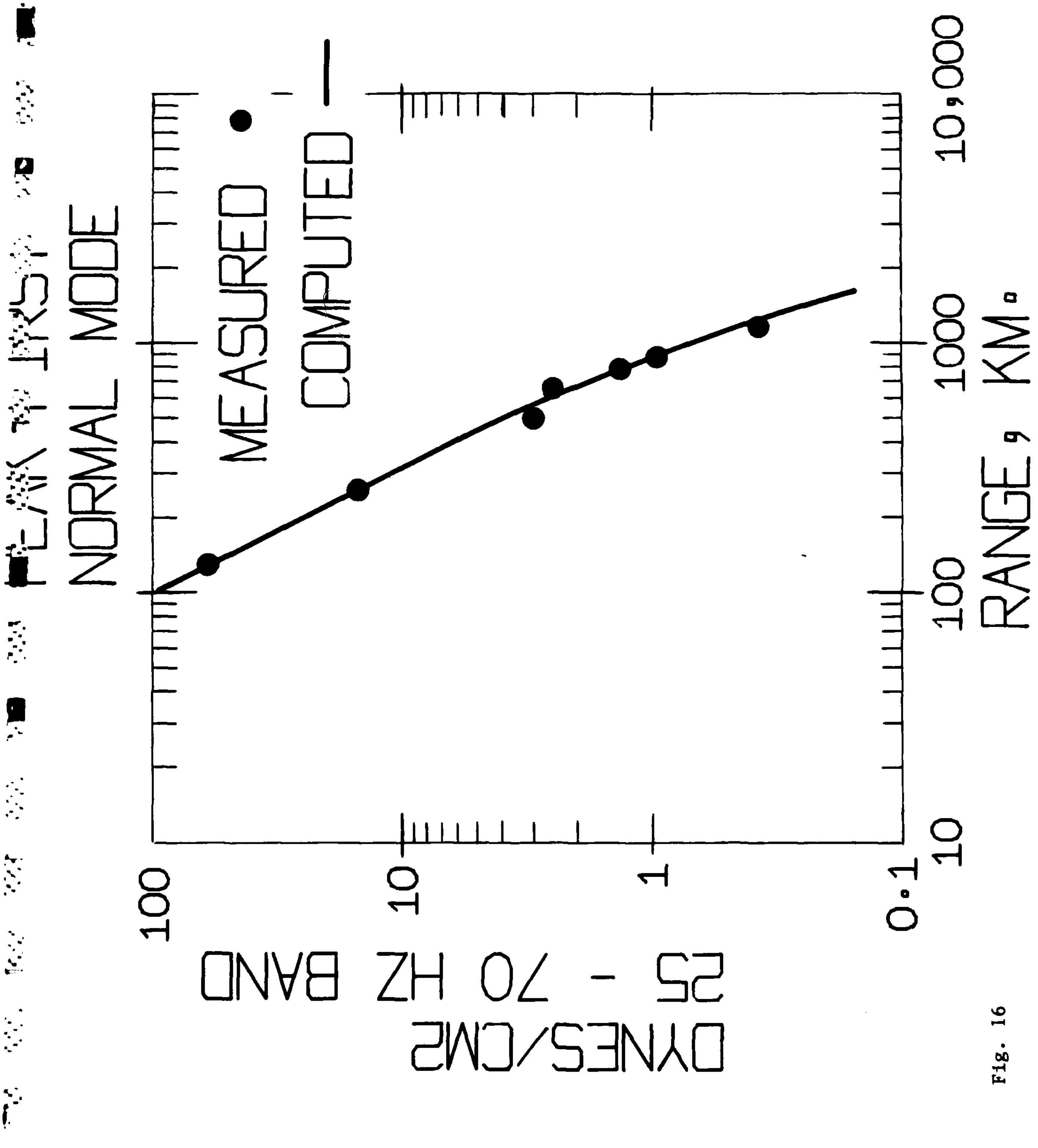
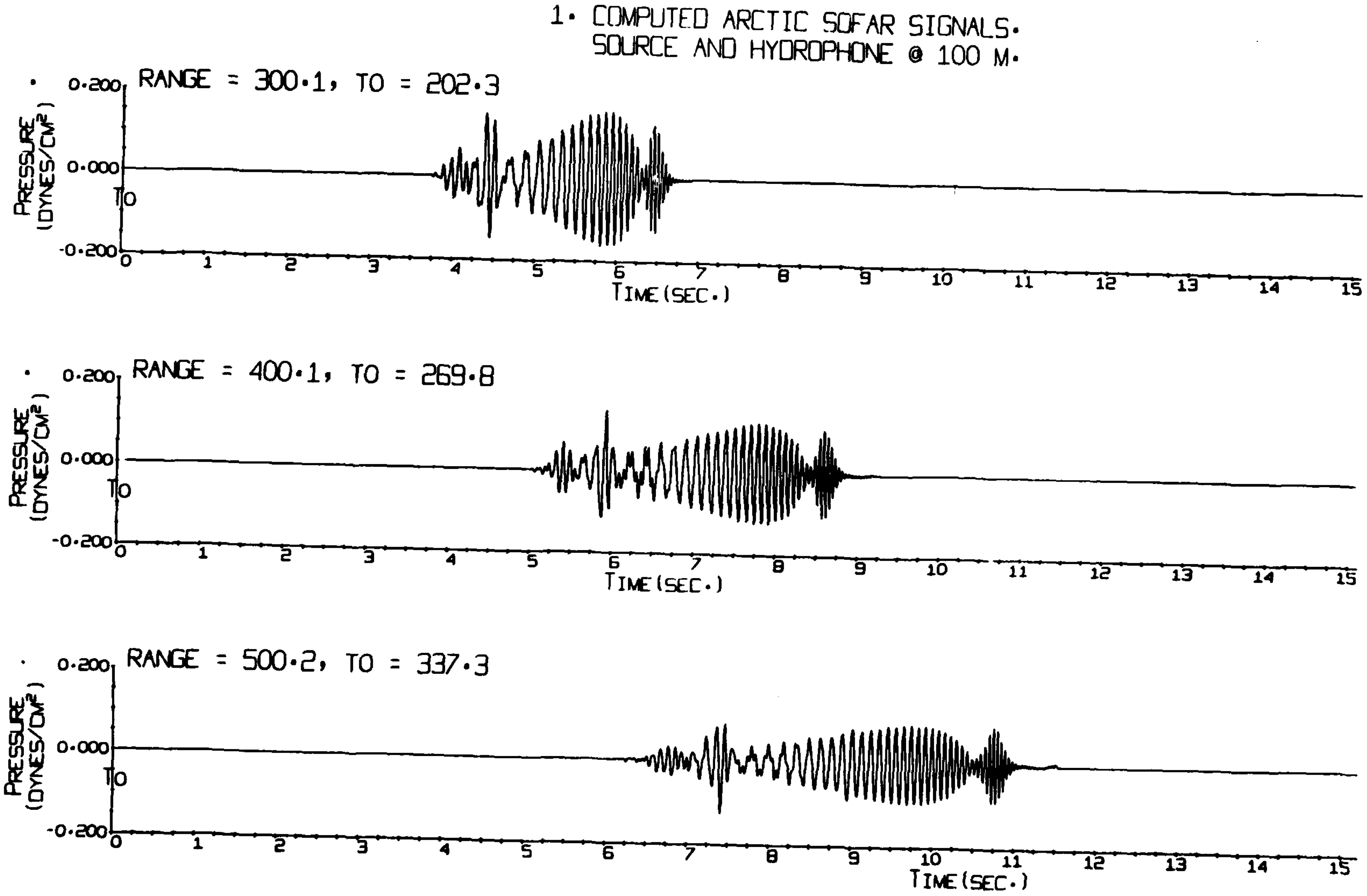
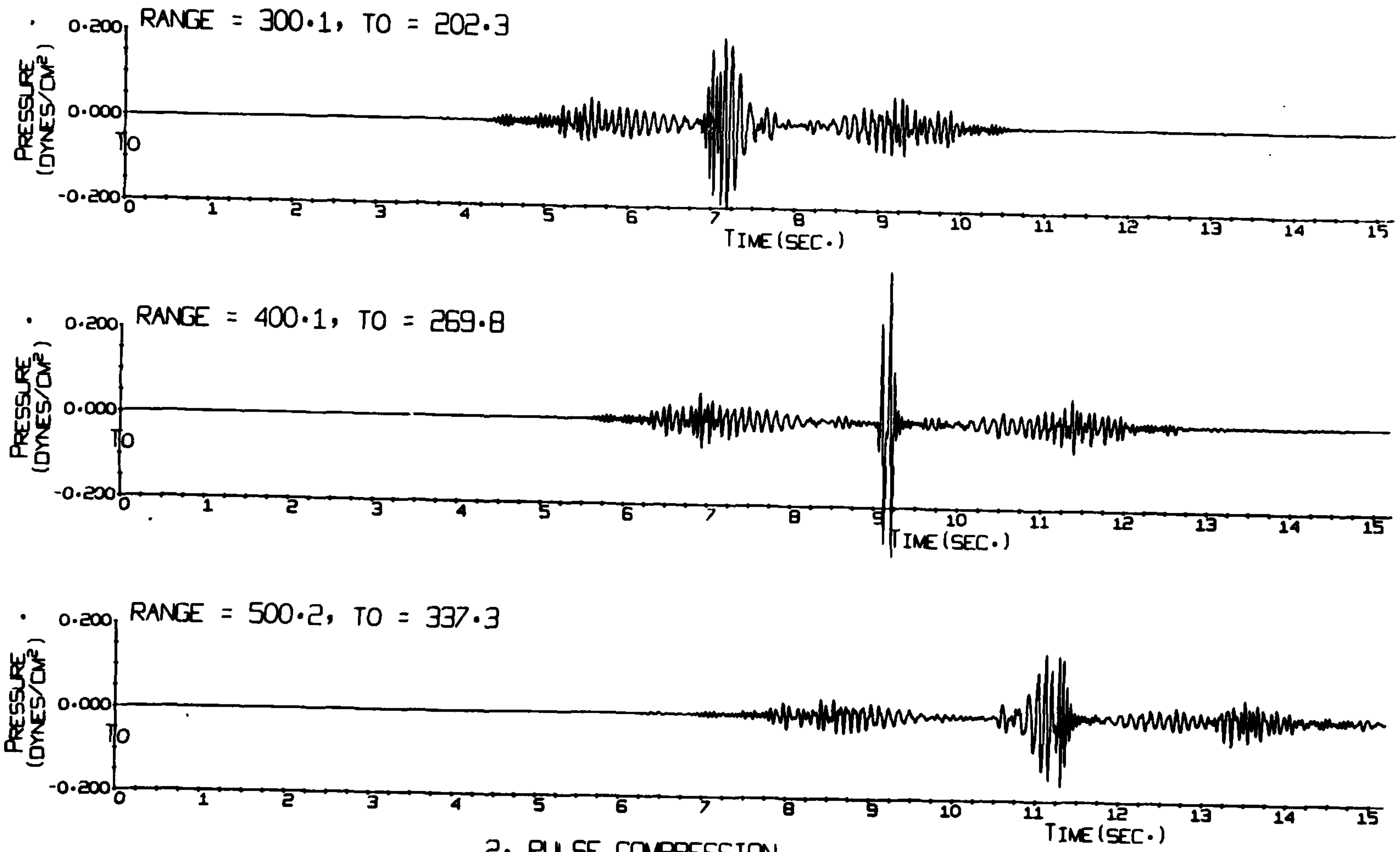

PEAK FIRST
NORMAL MODE
MEASURED ●
COMPUTED —
DYNES/CM2
25 - 70 HZ BAND
100
10
1
0.1
10
100
1000
10,000
RANGE, KM.

Fig. 16

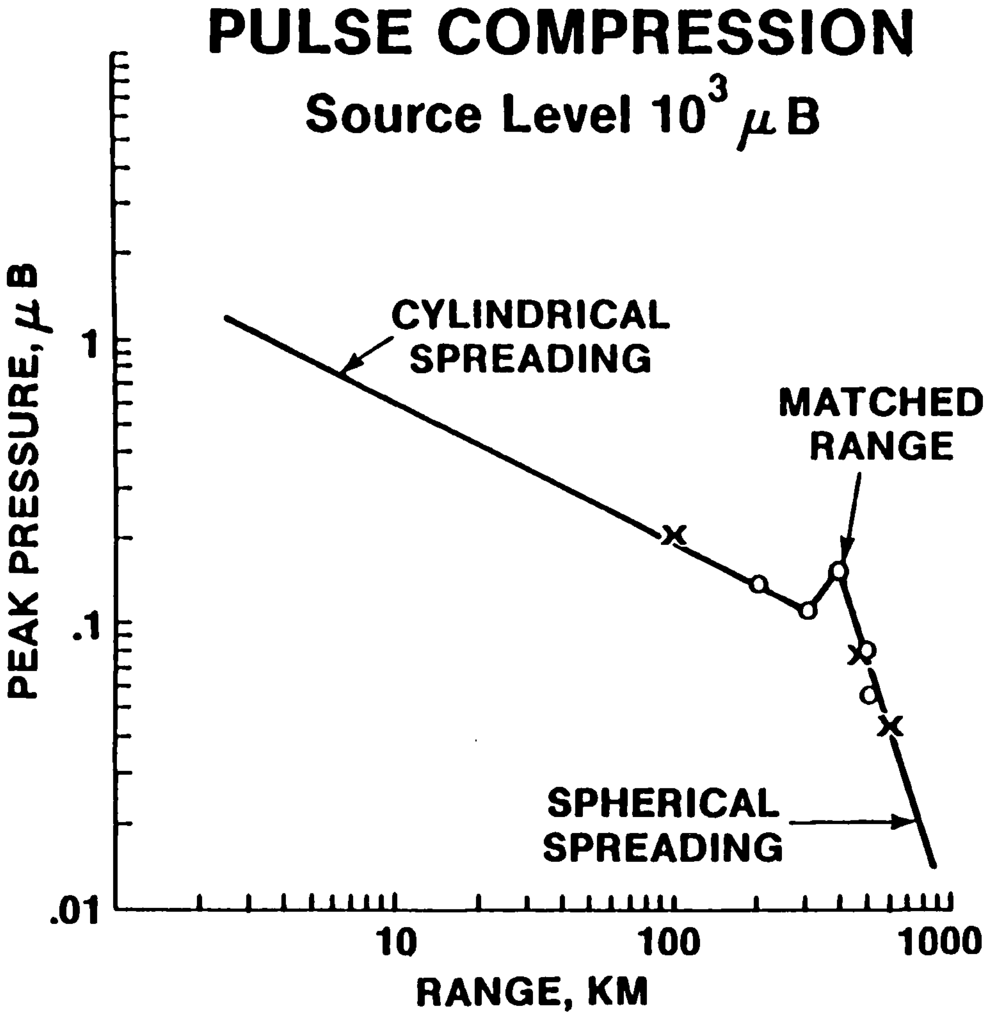
Fig. 17


1. COMPUTED ARCTIC SOFAR SIGNALS.
SOURCE AND HYDROPHONE @ 100 M.
RANGE = 300.1, TO = 202.3
RANGE = 400.1, TO = 269.8
RANGE = 500.2, TO = 337.3
PRESSURE (DYNES/CM^2)
0.200
0.000
-0.200
TO
0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15
TIME (SEC.)



2· PULSE COMPRESSION·
PREDISTORTED SIGNAL TIME REVERSE
OF 400 KM· SIGNAL OF FIG· 17
SOURCE AND HYDROPHONE @ 100 M·

Fig. 18



PULSE COMPRESSION

Source Level $10^3 \mu B$

Fig. 19

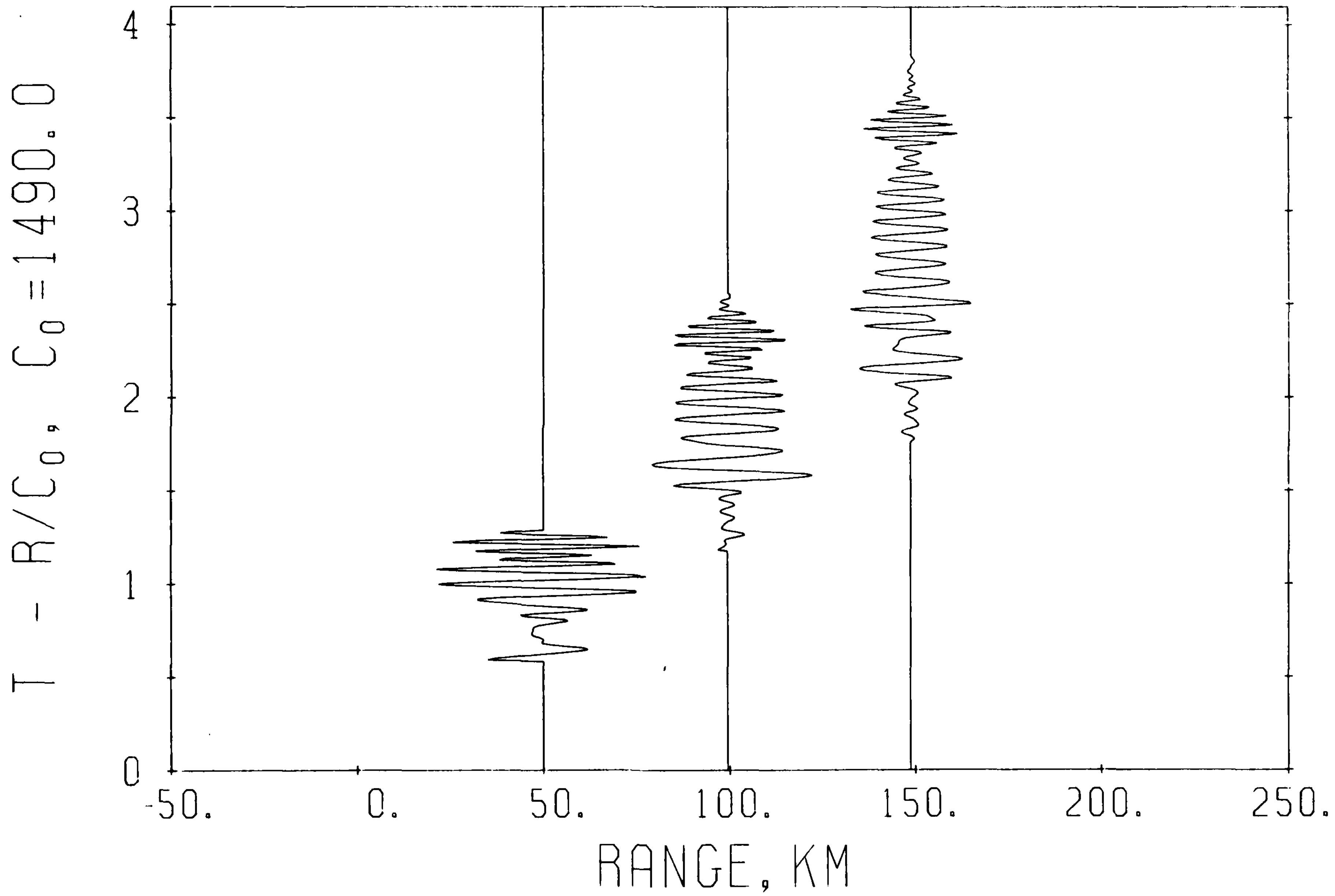


Fig. 1A

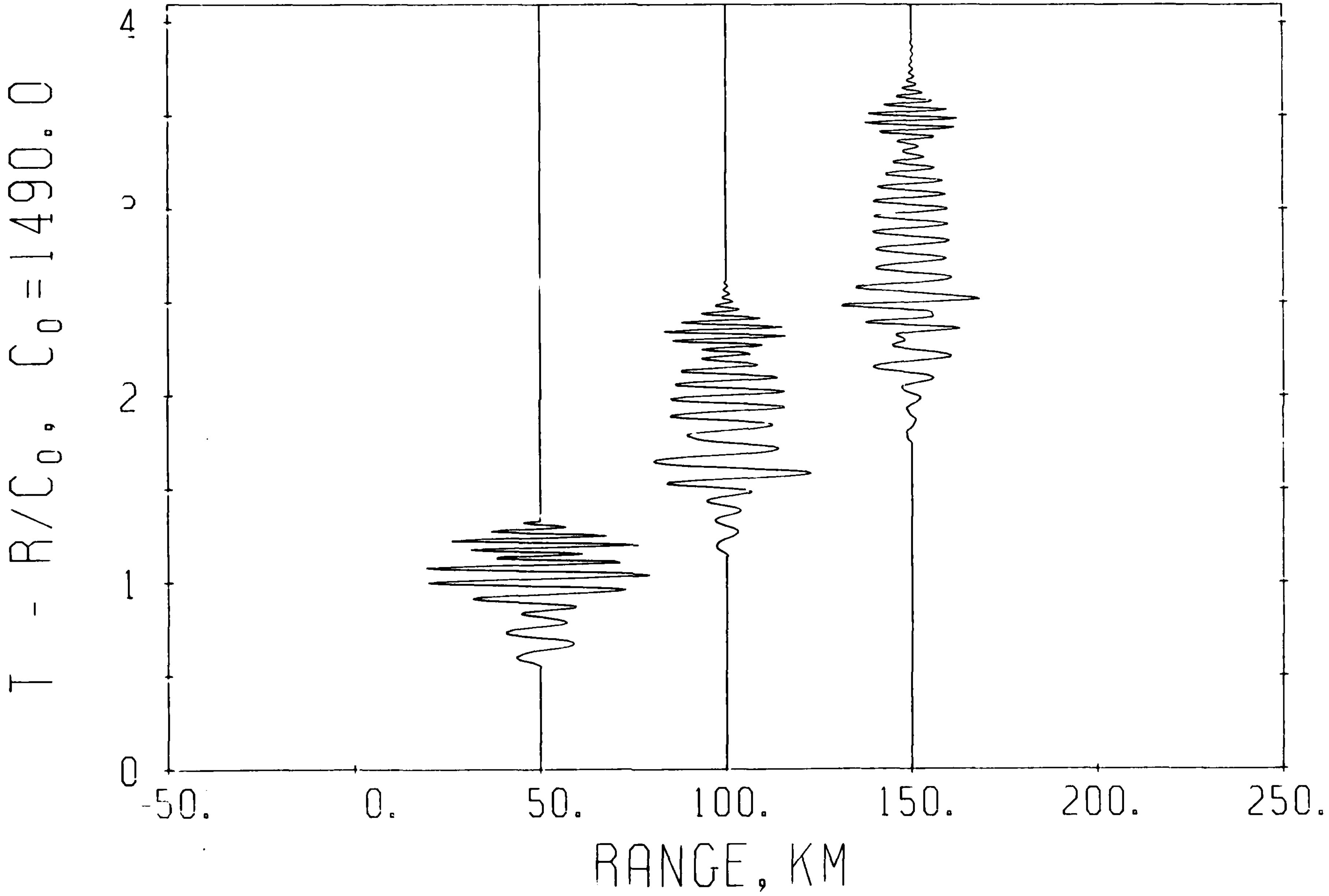


Fig. 2A

Invited Presentation

BOTTOM-INTERACTING ACOUSTIC SIGNALS IN THE ARCTIC CHANNEL:
LONG-RANGE PROPAGATION

Henry W. Kutschale and Tai Lee

Lamont-Doherty Geological Observatory
of Columbia University
Palisades, New York 10964

Presented at the 106th Meeting of the Acoustical Society of America,

San Diego, California

November 7 - 11, 1983

Abstract

Hydroacoustic signals from underwater explosions that have propagated over the Arctic abyssal plains commonly display marked frequency dispersion in pulses that are bottom-interacting and that arrive after the SOFAR signal. In the infrasonic band of 2 to 20 Hz, the temporal dispersion for each pulse that has interacted with the flat bottom of the plain can be nearly as strong as that observed in the SOFAR signal for the first mode. However, the bottom-interacting pulses correspond to a coherent summation of many higher-order normal modes in a channel bounded above by the ocean surface and below by the upper 400 m of the bottom sediments, where the velocity increases with depth. Using normal-mode theory and the Multiple Scattering Pulse Fast Field Program (MSPFFP), we have analyzed the dispersion and pulse shapes and have derived the acoustic properties of the bottom in the Pole, Barents, and Mendeleyev Abyssal Plains. The principal properties of the bottom controlling the propagation are compressional velocity, density, and attenuation. In contrast, the ice layer has a negligible effect on the dispersion of the observed waves. The effect on pulse compression of this frequency dispersion of the bottom-interacting signals was simulated numerically, using predistorted waveforms matched to the dispersion of the SOFAR channel at specified ranges.

Long-range transmissions of explosion sounds over the Arctic abyssal plains commonly show marked frequency dispersion in bottom-interacting pulses that arrive after the SOFAR signal[1]. Two types of frequency dispersion are observed. First, the frequency content of each successive bottom bounce is observed to decrease in the arrival sequence. The central frequency of each bottom-interacting pulse is dispersed from about 25 Hz at low-order bottom bounces to about 8 Hz at high-order bottom bounces. Second, we observe the dispersion in frequency of individual bottom-refracted pulses. These bottom-interacting pulses, which will be shown to have part of their travel paths through the bottom sediment, are markedly distorted in the frequency band from 5-20 Hz.

Previous investigations have revealed the unique characteristics of the Arctic SOFAR signal[2]. Fig. 1 shows a typical sound velocity profile in the Arctic channel. Fig. 2 shows that sound waves are transmitted long distances in this channel following successive RSR paths.

Shot signals in the Arctic channel clearly show low-order normal modes. The Arctic SOFAR signals are commonly nearly sinusoidal. This effect is caused by the constructive interference of RSR rays that have traveled in the upper few hundred meters of water, where the sound velocity increases rapidly with depth. Usually, at long-ranges, it is the first two normal modes that are observed in the frequency band below 40 Hz. However, Fig. 3 shows that if the channel is deep enough along the entire propagation path, such as over an abyssal plain, the early part of the arriving SOFAR wave train displays a discrete and impulsive character corresponding to the arrivals of deep-penetrating RSR rays, while the latter part of the wave train maintains its nearly sinusoidal character.

The dispersive character of the Arctic SOFAR signal is shown in Fig. 4, which is a sound spectrogram of the signal shown in Fig. 3. The dispersion of waves corresponding to the first and second normal modes is clearly visible in the frequency band below 40 Hz. The impulsive character of the deep-penetrating RSR waves is displayed by horizontal bands across the spectrogram, corresponding to a superposition of many high-order modes with group velocities equal to the mean horizontal velocities of each of the RSR pulses. Fig. 5 shows a comparison between the observed and computed dispersion of the first two normal modes. Agreement between observation and theory is excellent. The group-velocity dispersion of low-order modes in the Arctic channel is very uniform over a long propagation path, but the arrivals of the deep-penetrating RSR waves vary with range in both amplitude and mean horizontal velocity according to their cycle ranges. Because of scattering at the rough boundaries of the ice, only low frequencies can propagate to long distances in the Arctic channel.

An interesting observation in experiments of long-range acoustic transmission over the abyssal plains is the marked frequency dispersion of the bottom-interacting pulses that arrived after the SOFAR signal. This is shown in Fig. 6 for a typical signal propagating along a 316 km path over the Pole Abyssal Plain in a water depth of 4200 m. In Fig. 6 the temporal dispersion of the SOFAR signal is shown clearly. There are seventeen bottom-interacting pulses corresponding to the bottom bounces of order 11 to order 27 observed in Fig. 6 that arrived after the SOFAR signal. The total duration of the SOFAR signal and these bottom-interacting pulses is nearly fifty seconds. The bottom-interacting pulses arriving just after the SOFAR signal are very impulsive in character, but the bottom-interacting signals near the end of the wave train are nearly sinusoidal. Fig. 7 shows in detail the waveforms of two of the bottom-interacting pulses from Fig. 6. The pulses are normally dispersed with low-frequency waves traveling faster than high-frequency waves and the temporal dispersion of these bottom-interacting pulses is as strong as that of the first mode of the SOFAR signal.

The observed long-range bottom-interacting pulses may be resolved by ray acoustics into the multipath arrivals scattered from the top and the bottom boundaries of the Arctic channel. Thus they correspond to the summation of many high-order normal modes as do the deep-penetrating RSR waves. However, the frequency dispersion of individual bottom-interacting pulses is nearly as strong as that obseved in the SOFAR signal for the first mode. These dispersed bottom-interacting pulses must have traveled through and been distorted in the bottom sediment. This portion of the waveguide must be very uniform and have a dimension comparable to that of the channel in the upper few hundred meters of water, such that it can support the propagation of sound waves to long distances with frequencies comparable to that of the SOFAR signal.

The frequency dispersion of low-order normal modes has often been used to derive the acoustic properties of sediment beneath a shallow ocean waveguide since normal mode theory was developed by Pekeris[3,4]. In the present work, efforts have been made to explain the propagation in the Arctic channel of dispersed bottom-interacting pulses and to derive the acoustic properties of the bottom sediment from the dispersion of high-order normal modes. The techniques used in this study include normal-mode theory and the Multiple Scattering Pulse Fast Field Program (MSPFFP). Normal-mode theory is used to explain the frequency dispersion of the bottom-interacting wave trains. Fig. 8 shows that, although the dispersion of individual high-order modes cannot be resolved in the bottom-interacting wave train, the trend of the frequency reduction for bands present in the bottom-interacting sequence is found to correlate well with dispersion profiles of very high-order normal modes. Mode numbers as high as 60 may be observed in bottom-interacting signals with frequencies lower than 25 Hz. The MSPFFP is used to derive a geoacoustic model that predicts synthetic waveforms that match well with the dispersion profiles of the bottom interacting signals. The effect of the ice cover and shear waves in the sediment was investigated, and it was found that these two factors can be neglected in the model.

In general, the bottom-scattered wave trains show two types of pulses: high-frequency discrete arrivals and low-frequency, strongly distorted arrivals. The high-frequency discrete arrivals correspond to the post critical bottom-reflected pulses, while the low-frequency, strongly distorted arrivals correspond to the bottom-refracted pulses. Because of attenuation in the bottom-sediment, the bottom-refracted pulses contain lower frequencies than the bottom-reflected pulses.

The high-frequency very impulsive arrivals after the SOFAR signal are the bottom-reflected signals. Beyond a certain number of bottom bounces, the arrival of the low-frequency, distorted signals verifies the arrival of the bottom-refracted signals. In the bottom-refracted signals, two types of pulse distortion can be found on the spectrogram of Fig. 8. First, the pulses are normally dispersed corresponding to multipath propgation of pulses through sediment with a positive velocity gradient. Second, after the arrival of the normally-dispersed waves, pulses arrive which show a branch of inverse frequency dispersion. These bottom-scattered waves are still bottom-refracted waves, but they have penetrated into a deeper sediment layer with a different velocity gradient. Therefore, the prototype model which supports the propagation of the observed bottom-interacting signals consists of three parts:

the ocean, the shallow sediment with a strong velocity gradient and the deep sediment with a weaker velocity gradient.

The Multiple Scattering Pulse Fast Field Program decomposes the Pulse Fast Field Program[5,6] into ray path type contributions[7]. Each decomposed term can be interpreted as the desired path contribution for a corresponding bottom-interacting pulse. The Fast Field Program (FFP) algorithm integrates directly the full wave solution. Thus, the MSPFFP technique is a natural scheme with which to model the bottom-interacting pulses, which correspond to the coherent summation of many modes over a limited time interval. Temporal waveforms are computed by Fourier synthesis.

We started our analysis by using a simple Pekeris model to identify the order of the bottom-scattered waves. Then we applied a method of critical reflections and refractions used by Katz and Ewing[8] to derive a preliminary geoacoustic model. This geoacoustic model was refined by matching the dispersion of the theoretical bottom-refracted waves to the dispersion of the observed bottom-refracted waves.

In Fig. 9 the observed and theoretical reduced travel-time curves for shots recorded from two seismic refraction lines during the FRAM II Expedition are shown. The triangle mark in Fig. 9 represents the observed reduced travel time at the onset of each bottom bounce. In general, the theoretical travel time curves are in good agreement with the observed ones. This is not to say that this simple model, a two-layered Pekeris waveguide, is good enough to explain the propagation of the bottom-interacting signals in the Arctic ocean; the theoretical travel time curves are the travel time curves of the bottom-reflected signals only. The bottom-interacting pulses of order more than 17 in Fig. 8 are bottom-refracted pulses with part of their travel paths through the bottom sediment. However, the good agreement between the theoretical and observed travel-time curves may verify the identification of the order of the bottom-scattered waves.

For shots recorded at close ranges, the number of bottom bounces is easy to count, but for those shots recorded at long ranges, the order of bottom bounces is not as easily identified. The reduced travel time versus distance plot offers us a convenient method with which to identify the orders of the arriving pulses. Bottom-scattered waves up to twelfth order were observed for 100 km shots at FRAM II and bottom-scattered waves of more than thirty-second order were observed for 340 km shots at Camp I.

On the basis of bottom-reflections at the critical angle, the sound velocity in sediments near the ocean floor can be calculated form the maximum number of bottom reflections[8].

Assuming that the curvature of the reflected ray in the water column is negligible and that the maximum number of bottom reflections on a given record is determined by the critical angle of reflection, the approximate value of the sound velocity beneath the sea floor calculated by this method is 1.63 km/sec. The sound velocity at the bottom of the Arctic Ocean, 4.2 km in depth, is about 1.520 km/sec. The estimated sound velocity beneath the sea floor of 1.63 km/sec does not firmly suggest that a first-order discontinuity exists at the bottom interface. A thin layer with a strong velocity gradient is more likely to represent the real situation.

Assuming a constant positive velocity gradient in the shallow sediment, the velocity gradient and the depth of deepest penetration can be estimated from the cycle range, travel time and phase velocities. The phase velocities are estimated from the theoretical travel time curves shown in Fig. 9. These computations agree well with a model having a velocity gradient of 2.0 s^{-1}. The thickness of the shallow sediment as defined by this velocity gradient is about 100 meters. The same method can be applied to derive the velocity gradient in the deeper sediment layer. Repeating the same procedure, a velocity gradient of 1.0 s^{-1} is derived.

From this estimate of the velocity gradients in the sediment and from the analysis of the pulse shapes of the bottom-interacting signals, a geoacoustic model for the wave propagation in the Pole Abyssal Plain may be derived. Fig. 10 shows the velocity profile of this acoustic model, and Fig. 11 shows the density profile. This velocity profile extends the Arctic SOFAR channel several hundred meters into the bottom sediment.

Fig. 12 shows an observed wave train recorded at a range of 295 km. The signal was recorded by a hydrophone 50 meters deep.

In Fig. 12, only seven of the bottom-interacting pulses in the computed wave train are shown. This is to indicate the ability of the Multiple Scattering Pulse Fast Field Program to perform the computation for a single pulse, such as those bottom multiples of order 14, 16, 18 and 20, or for several pulses at one time, such as the bottom multiples of order 22 to 24. One can start by modeling the lower order bottom bounces (such as M18 or M20, which travel through the shallow sediment layer) in order to derive the structure of the shallow sediment, then extend the structure deeper to model the high-order bottom bounces. This procedure shows the advantage of the Multiple Scattering Pulse Fast Field Program. With the MSPFFP, one can calculate one or more bottom-interacting pulses, testing and modifying the model, rather than compute the whole wave train. This saves a great deal of computing time, especially when a wide range of model parameters has to be tested.

In Fig. 13 the observed SOFAR signal from Fig. 12 and a computed SOFAR signal are compared. In Fig. 14 the dispersion profiles of the observed and computed SOFAR signal are shown. These dispersion profiles were calculated by the conventional "graphical peak and trough" method and were checked by the spectral filtration method. The agreement in the dispersion profile for the first mode of the SOFAR signal is excellent. We used the same procedure to compare the dispersion profiles of the computed and observed bottom-interacting signals.

The next three figures, 15, 16 and 17, show the comparison of the observed and computed waveforms for the bottom bounces of orders 20, 22 and 24. These bottom-interacting signals are the same as those shown in Fig. 12. The dispersion profiles of the bottom-interacting signals having similar cycle ranges of 15 km, 13.5 km and 12.5 km are shown in the next three figures, 18, 19 and 20. If the distance traveled in the sediment is long enough such that constructive interference of rays in the sediment has occurred, then the bottom-interacting signals having the same cycle range (or the same phase velocity) should have nearly the same dispersion profiles. As shown in these figures, the dispersion profiles of the computed waveforms and those of the observed waveforms with about the same cycle ranges are in good agreement.

The agreement between the dispersion profiles of shot signals at different ranges suggests that a very uniform sediment waveguide exists below the Pole Abyssal Plain in the Arctic Ocean. The constructive interference of rays having a travel path through the sediment waveguide is responsible for the observed frequency dispersion of the individual bottom-interacting pulses. The agreement of the dispersion profiles for the computed and observed bottom-interacting pulses suggests that the geoacoustic model is a fairly good model for the sediment waveguide below the Pole Abyssal Plain. In this geoacoustic model, a positive acoustic wave velocity gradient decreases with depth from about 2 s $^{-1}$ at the sediment surface to about 1.0 s^{-1} at about 400-500 meters depth which is, in general, consistent with observations in the other oceans.

The existence of a low velocity layer in the superficial layer of the deep-sea sediment has been observed, but because the frequency content of the bottom-interacting signals is less than 40 Hz, the existence of a very thin low-velocity layer cannot be resolved from our data. However, in the numerical simulations of high-order bottom-interacting signals, a layer of low-velocity sediment 15 m thick is used because of better agreement in the high-frequency portion of the 24th bottom bounce than that computed without a low-velocity layer. But, in general, there are no differences in the frequency dispersion of these two waveforms.

It can be concluded that a uniform waveguide exists below the Pole Abyssal Plain. The consistency of the observed signals suggests that the bottom sediment waveguide may be considered a part of the Arctic SOFAR channel. The velocity profile in the surface channel is responsibile for the frequency dispersion of the Arctic SOFAR signals, and the positive velocity gradient in the bottom sediment is responsible for the dispersion of the bottom-refracted pulses. For long-range sound propagation over the Arctic abyssal plains, at least up to 400 km in range, the bottom sediment structure several hundred meters below the sea floor is important.

The geoacoustic model derived from Pole Abyssal Plain was used to model the bottom-interacting signals recorded in both the Mendeleyev Abyssal Plain and the Barents Abyssal Plain. Good agreement in wave forms of the observed and computed bottom-interacting signals suggests that similar bottom structure may exist below these abyssal plains of the Arctic Ocean, at least for the upper few hundred meters. The geoacoustic model derived from the Pole Abyssal Plain is also similar to that proposed by Hamilton[9]. Although the depth-dependent attenuation coefficients could not be resolved accurately from our data, our average value of 0.8 dB/km at 10 Hz is in the range of the attenuation coefficients given by Hamilton.

The uniformity of group-velocity dispersion of waves corresponding to the first normal mode in the frequency band 8 to 30 Hz suggested the possibility of achieving significant signal gain by pulse compression in the Arctic SOFAR channel[10]. Numerical simulations shown in Fig. 21 were accomplished by propagating time reversed predistorted wave forms matched to the dispersion of the channel at specified ranges. The similarity of the frequency dispersion between the Arctic SOFAR signals and the bottom-interacting signals over the abyssal plains suggests the possibility of interference with a pulse compression scheme.

A computed shot signal, including all the bottom-interacting pulses, is shown in Fig. 22. Fig. 23 shows a simulation of pulse compression by cross-correlating the SOFAR signal and the total response including all the bottom interactions. Since the dispersion of the bottom-refracted pulses is similar to that of the SOFAR signals, pulse compression of the bottom-interacting response by the SOFAR channel response is expected. It can be seen in Fig. 23 that after a travel time of 230 seconds, the bottom-refracted pulses are indeed compressed, especially those pulses (M20, M21) which deeply penetrate the shallow sediment waveguide (with a strong velocity gradient) but do not travel into the deeper sediment layer (with a weaker velocity gradient). Fig. 24 shows the details of the pulse compression of the bottom-interacting signals of order 18 to 24. The amplitude scale in Fig. 24 is three times that in Fig. 23. The most compressed pulse corresponds to the bottom bounce of order 21.

The effect of the bottom-interacting signals on pulse compression has been shown by computer simulation. The pulse compression of the SOFAR signal is observed, but many more sidelobes occur. The sidelobes generate more noise for the pulse compression scheme to deal with. The dispersed bottom-refracted pulses are indeed compressed during the simulation, but the amplitude of the compressed bottom refracted pulses is much smaller than that of the compressed SOFAR signal.

References

[1] H.W. Kutschale, "Pulse Propagation in the Central Arctic Ocean," J. Acoust. Soc. Am. Suppl. 1, 69, S19 (1981).

[2] H. Kutschale, "Long-Range Sound Transmission in the Arctic Ocean," J. Geophys. Res. 66, 2189-2198 (1961).

[3] C.L. Pekeris, "Theory of Propagation of Explosive Sound in Shallow Water," in Propagation of Sound in the Ocean, Geol. Soc. Am. Mem. 27, 1-117 (1948).

[4] J.L. Worzel and M. Ewing, "Explosion Sounds in Shallow Water," in Propagation of Sound in the Ocean, Geol. Soc. Am. Mem. 27, 1-53 (1948).

[5] H.W. Kutschale, "Rapid Computation by Wave Theory of Propagation Loss in the Arctic Ocean," Lamont-Doherty Geological Observ. Columbia U. Tech. Rep. CU-8-73 (1973).

[6] H.W. Kutschale and F.R. DiNapoli, "Pulse Propagation in the Ocean. II: The Fast Field Program Method," J. Acoust. Soc. Am. Suppl. 1, 62, S18 (1977).

[7] D.S. Ahluwalia and J.B. Keller, "Exact and Asymptotic Representations of the Sound Field in a Stratified Ocean," in Wave Propagation and Underwater Acoustic, edited by J.B. Keller and J.S. Papadakis, Lecture Notes in Physics 70, 14-85 (Springer, New York, 1977).

[8] S.Katz and M. Ewing, "Seismic-Refraction Measurements in the Atlantic Ocean Part VII: Atlantic Ocean Basin, West of Bermuda," Bull. Geol. Soc. Am. 67, 475-509 (1956).

[9] E.L. Hamilton, "Geoacoustic Modeling of the Sea Floor," J. Acoust. Soc. Am. 68, 1313-1340 (1980).

[10] F.R. DiNapoli, D. Viccione and H. Kutschale, "The Arctic Channel: An Acoustic Waveguide," Naval Underwater System Center Technical Memorandum No. 781078 (1978).

Captions for Figures

Fig. 1 Velocity profile in Eurasian Basin calculated from hydrographic data.

Fig. 2 Ray paths computed at one-degree interval.

Fig. 3 Waveform of typical SOFAR signal measured in Eurasian Basin.

Fig. 4 Sound spectrogram of typical SOFAR signal measured in Eurasian Basin. The signal is shown in Fig. 3. M1 first mode. M2 second mode.

Fig. 5 Comparison of measured and computed group-velocity dispersion for the first two normal modes.

Fig. 6 Measured waveform which shows strong bottom-interacting signals. Range 333.5 km.

Fig. 7 Waveforms of two bottom-interacting signals of Fig. 6.

Fig. 8 Waveform and corresponding sound spectrogram of typical signal measured over flat bottom of the Pole Abyssal Plain. Mean water depth 4200 m. Range 317 km. Charge size 55 pounds TNT. Dispersion profiles for the 17th and 29th mode (M17, M29) are shown on the spectrogram.

Fig. 9 Plot of the reduced travel time curves of shot records from two seismic refration lines of the FRAM II Expedition. The shots in the 0 - 80 km range were recorded at FRAM II and the shots in the 200 - 340 km range were recorded at Camp I. The continuous lines are theoretical reduced travel time curves computed from a two-layered Pekeris waveguide.

Fig. 10 The velocity profile of the geoacoustic model.

Fig. 11 The density profile of the geoacoustic model.

Fig. 12 Measured and computed waveforms which show strong bottom-interacting signals. Range 295 km. Charge size 880 pounds TNT. The computed waveforms were calculated by the Multiple Scattering Pulse Fast Field Program (MSPFFP).

Fig. 13 Measured and computed SOFAR signal.

Fig. 14 Dispersion profile corresponding to the first mode of the observed and computed SOFAR signal in Fig. 13.

Fig. 15 Observed and computed waveforms of 20-th bottom bounce (M20) in Fig. 12.

Fig. 16 Observed and computed waveforms of 22-nd bottom bounce (M22) in Fig. 12.

Fig. 17 Observed and computed waveforms of 24-th bottom bounce (M24) in Fig. 12.

Fig. 18 Dispersion profiles for bottom-interacting signals having a cycle range about 15 km. The dispersion profiles of waveforms shown in Fig. 15 are included.

Fig. 19 Dispersion profiles for bottom-interacting signals having a cycle range about 13.5 km. The dispersion profiles of waveforms shown in Fig. 16 are included.

Fig. 20 Dispersion profiles for bottom-interacting signals having a cycle range about 12.5 km. The dispersion profiles of waveforms shown in Fig. 17 are included.

Fig. 21 Computed SOFAR signals at ranges from 300 to 900 km and pulse compression of SOFAR signals. Source signal for pulse compression is the time reverse of second SOFAR signal from top.

Fig. 22 Computed waveform at 295 km, including all the bottom-interacting signals. Geoacoustic model of the bottom is shown in Figs. 10 and 11. The water depth is 4200 m.

Fig. 23 Pulse compression at a range of 295 km. All the bottom-interacting signals are included.

Fig. 24 The details of the pulse compression shown in Fig. 23 of bottom-interacting signals of order 18 to 24.

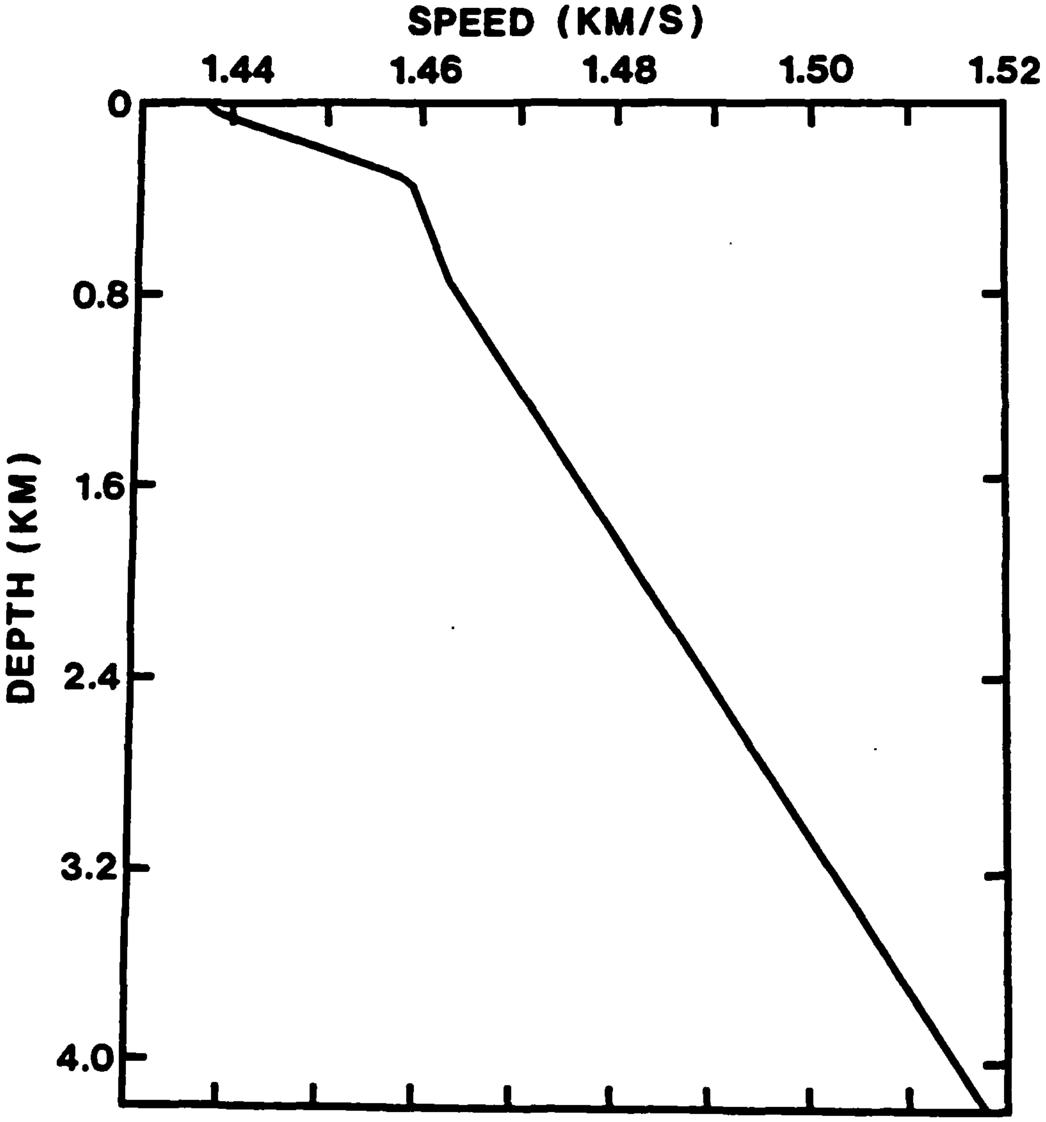


Fig. 1

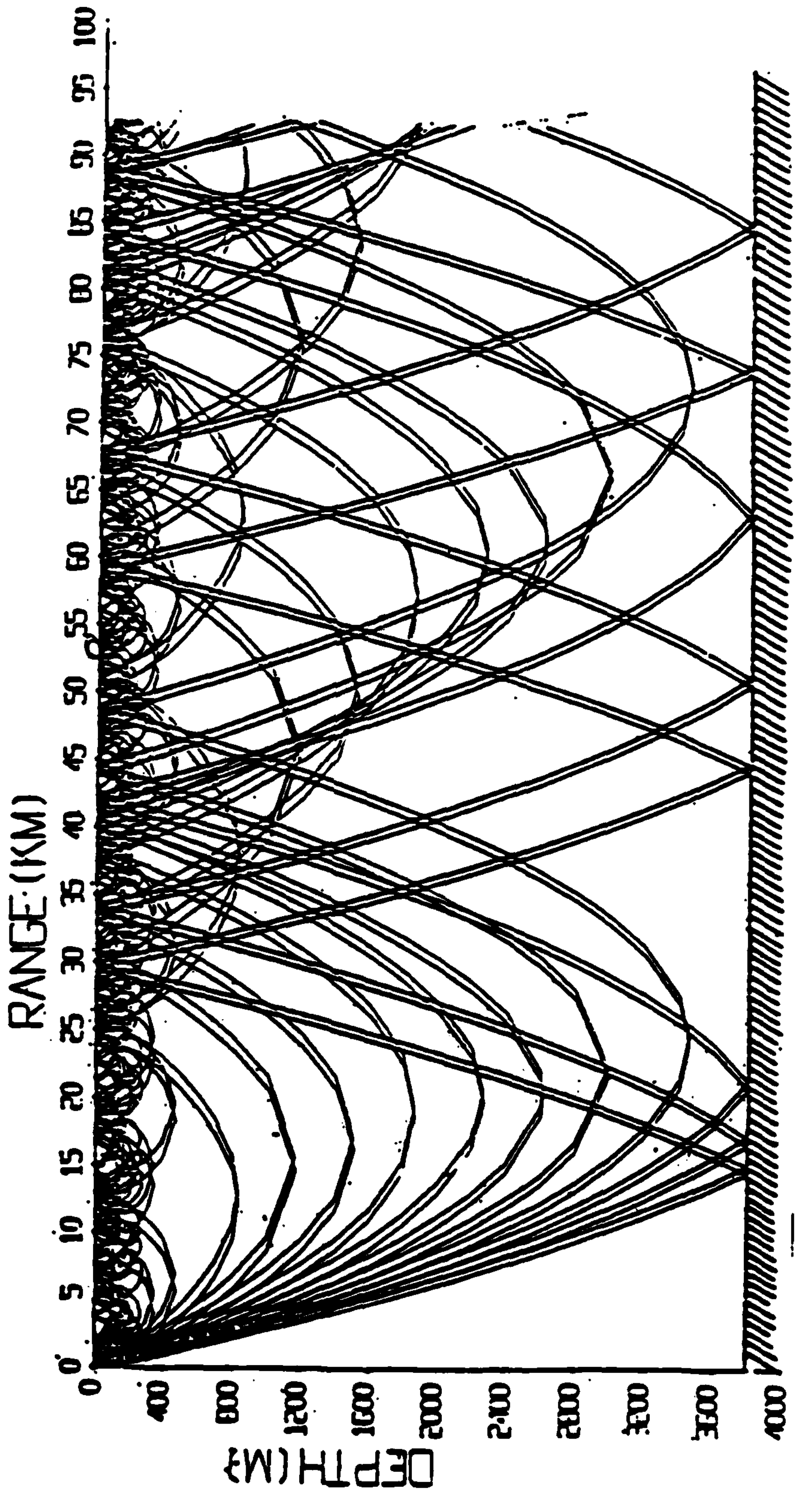

RANGE (KM)
0
5
10
15
20
25
30
35
40
45
50
55
60
65
70
75
80
85
90
95
100
DEPTH (M)
0
400
800
1200
1600
2000
2400
2800
3200
3600
4000

Fig. 2

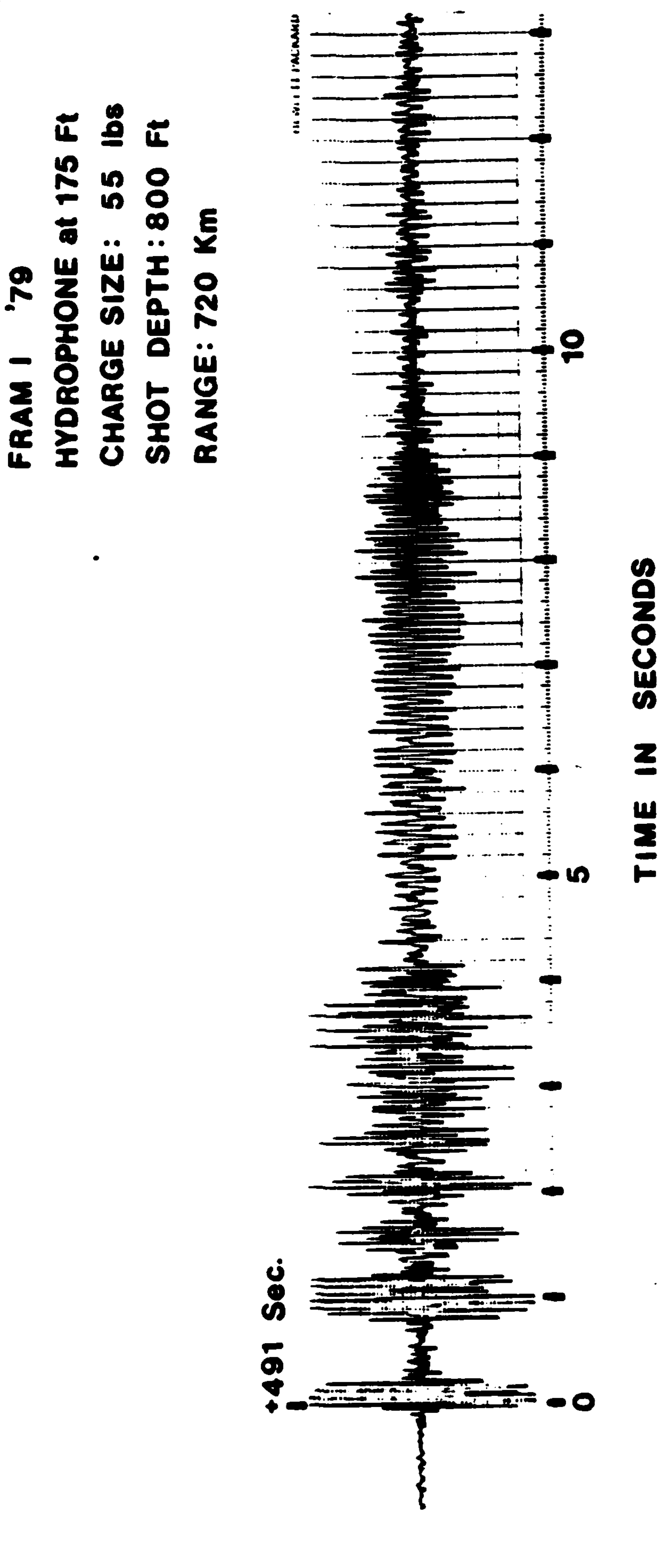

FRAM I '79
HYDROPHONE at 175 Ft
CHARGE SIZE: 55 lbs
SHOT DEPTH: 800 Ft
RANGE: 720 Km
+491 Sec.
0
5
10
TIME IN SECONDS

Fig. 3

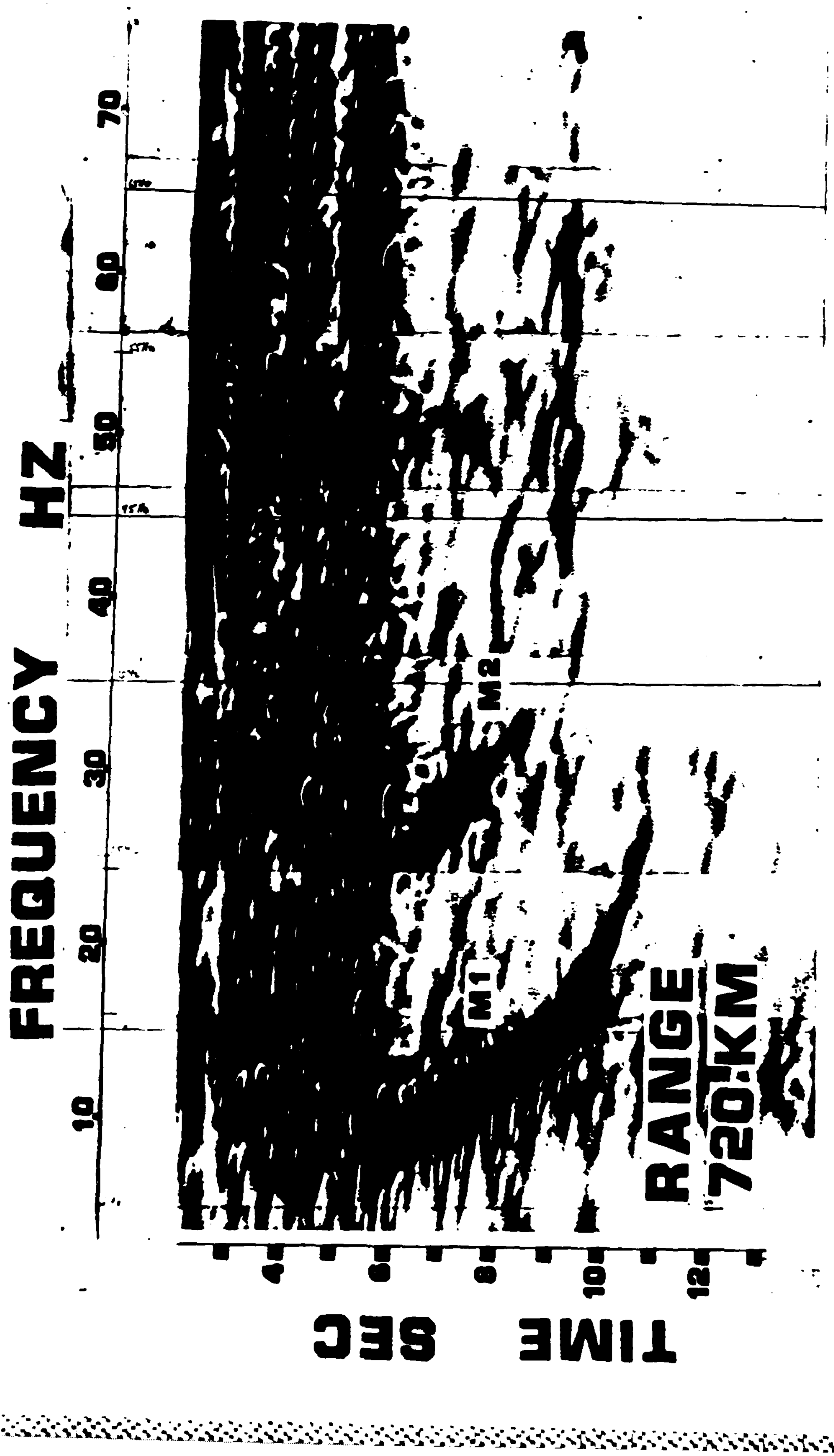

FREQUENCY HZ
10
20
30
40
50
60
70
M1
M2
RANGE
720 KM
TIME SEC
4
6
8
10
12

Fig. 4

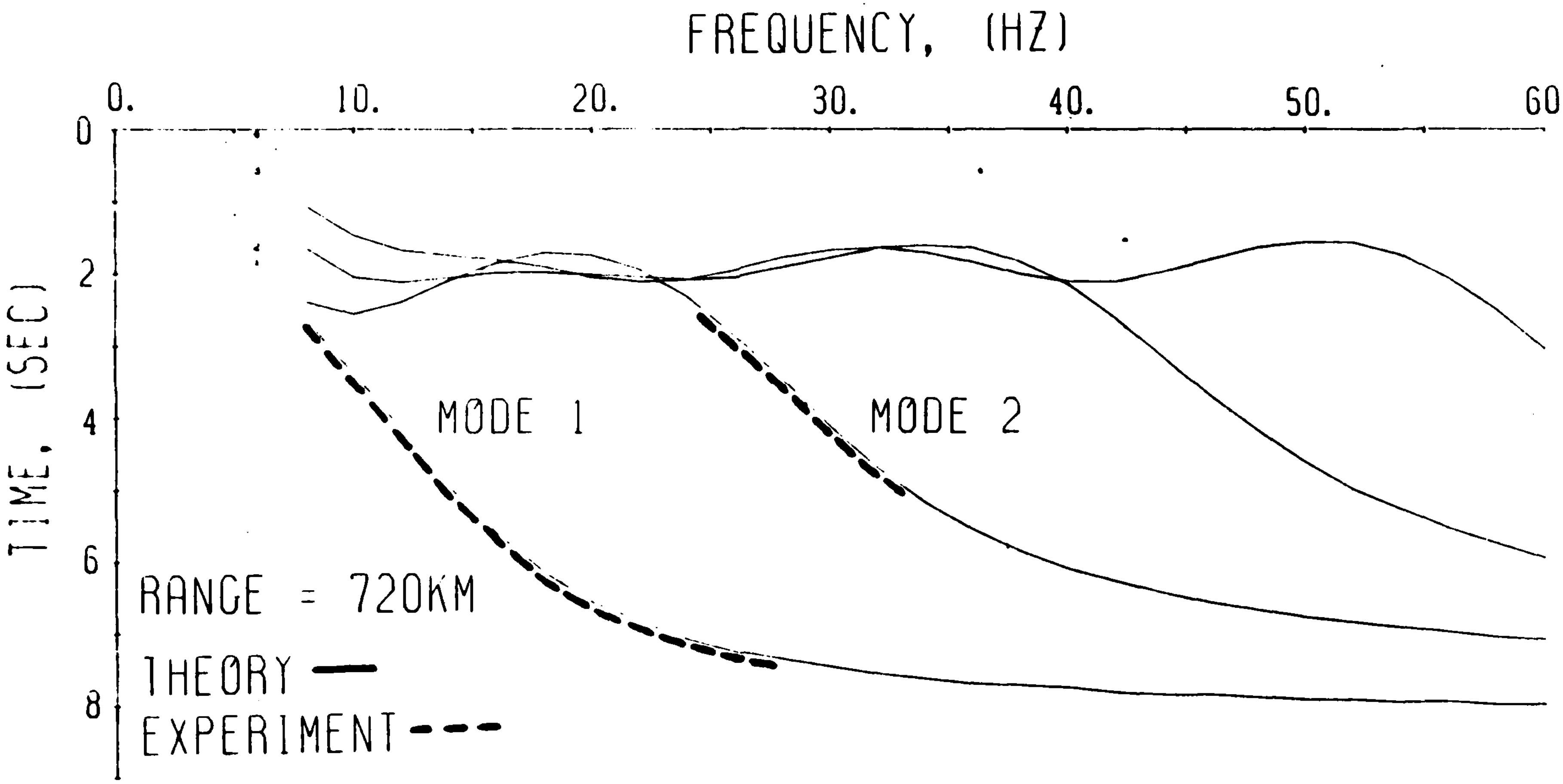

FREQUENCY, (HZ)
0.
10.
20.
30.
40.
50.
60.
0
2
4
6
8
TIME, (SEC)
MODE 1
MODE 2
RANGE = 720KM
THEORY
EXPERIMENT

Fig. 5

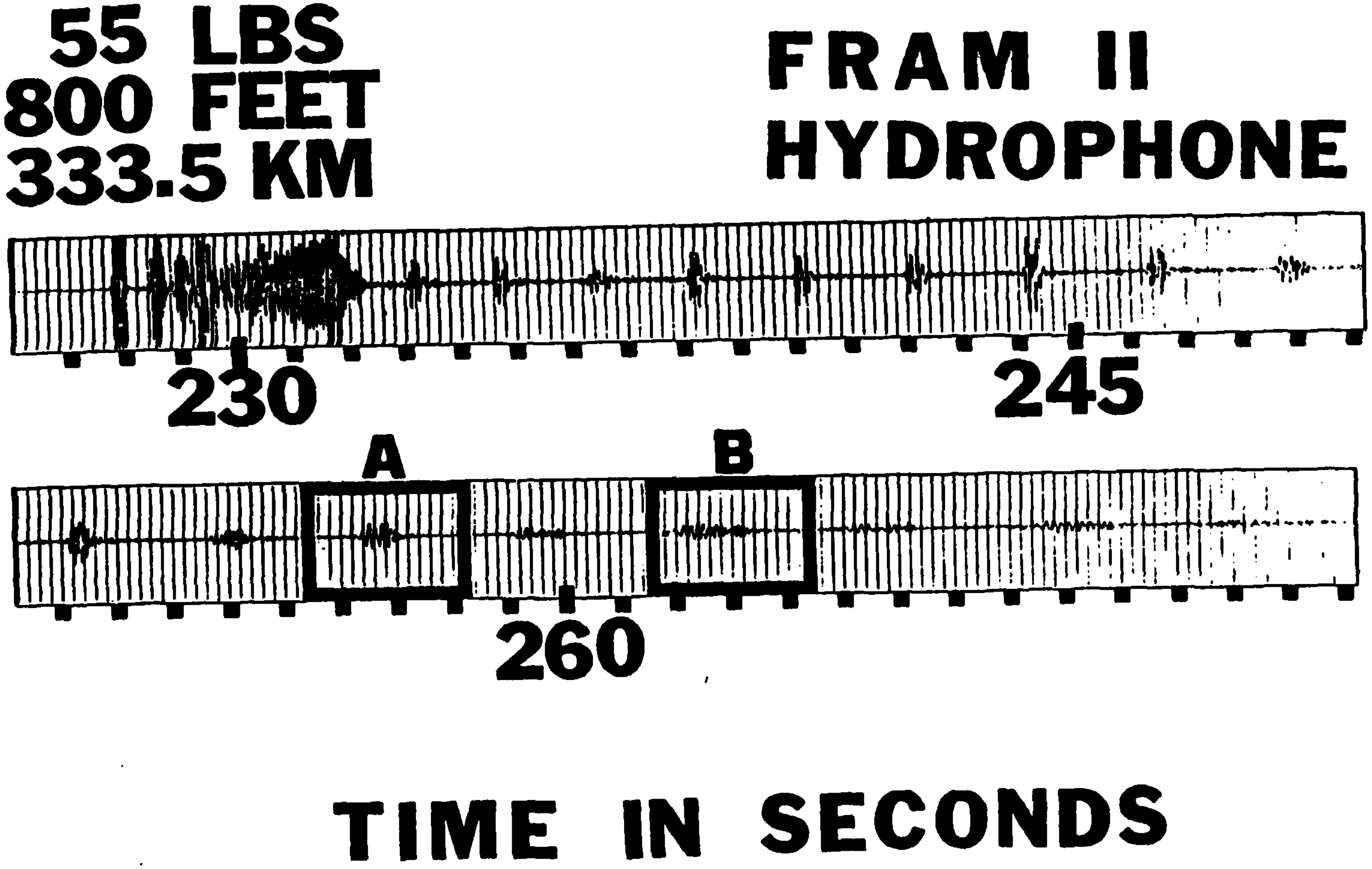

55 LBS
800 FEET
333.5 KM
FRAM II
HYDROPHONE
230
245
A
B
260
TIME IN SECONDS

Fig. 6

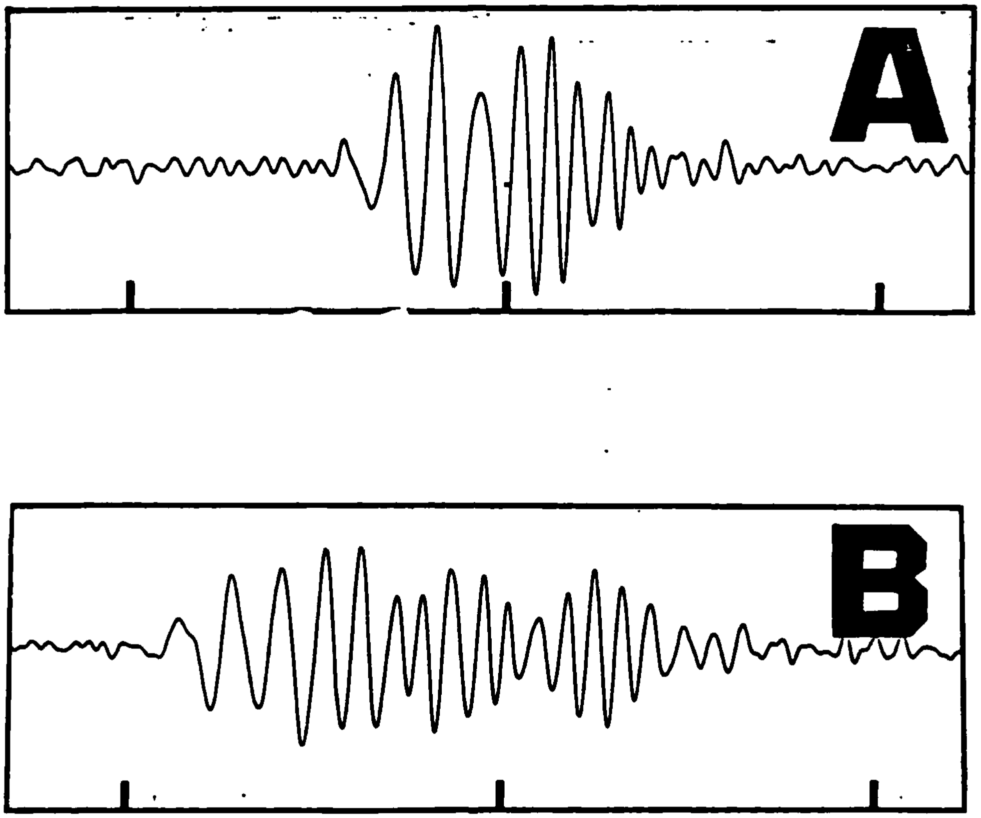


Fig. 7

RANGE 317 KM

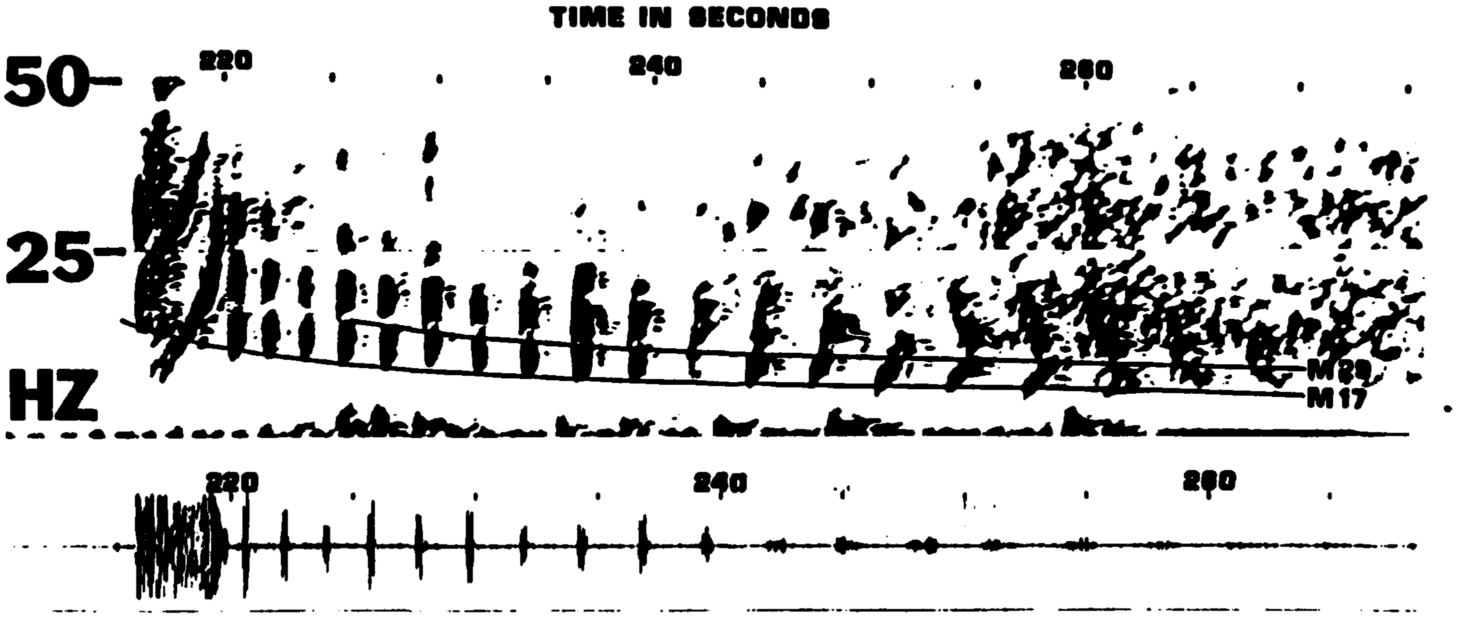


Fig. 8

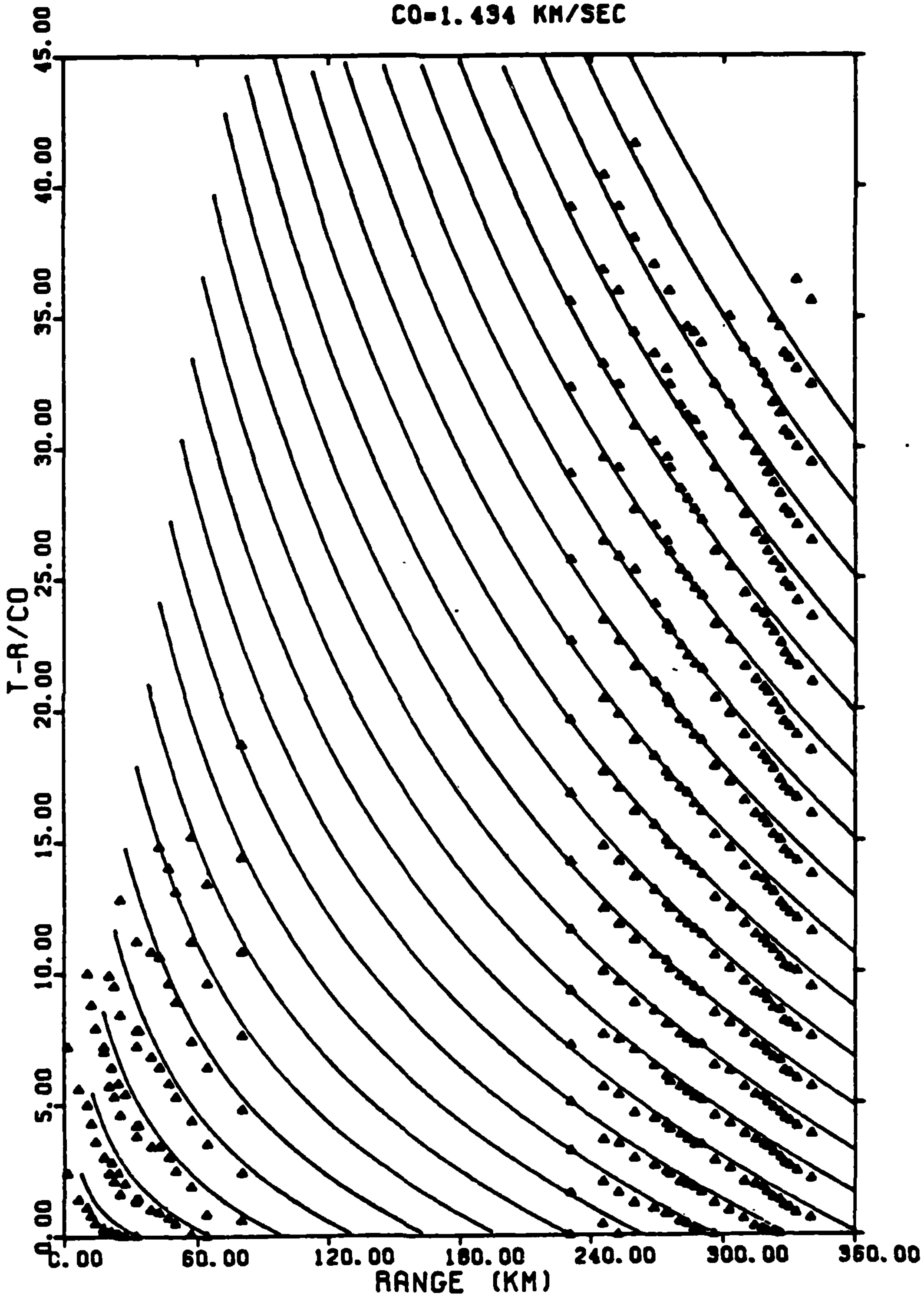

CO=1.434 KM/SEC
T-R/CO
RANGE (KM)
0.00
5.00
10.00
15.00
20.00
25.00
30.00
35.00
40.00
45.00
0.00
60.00
120.00
180.00
240.00
300.00
360.00

Fig. 9

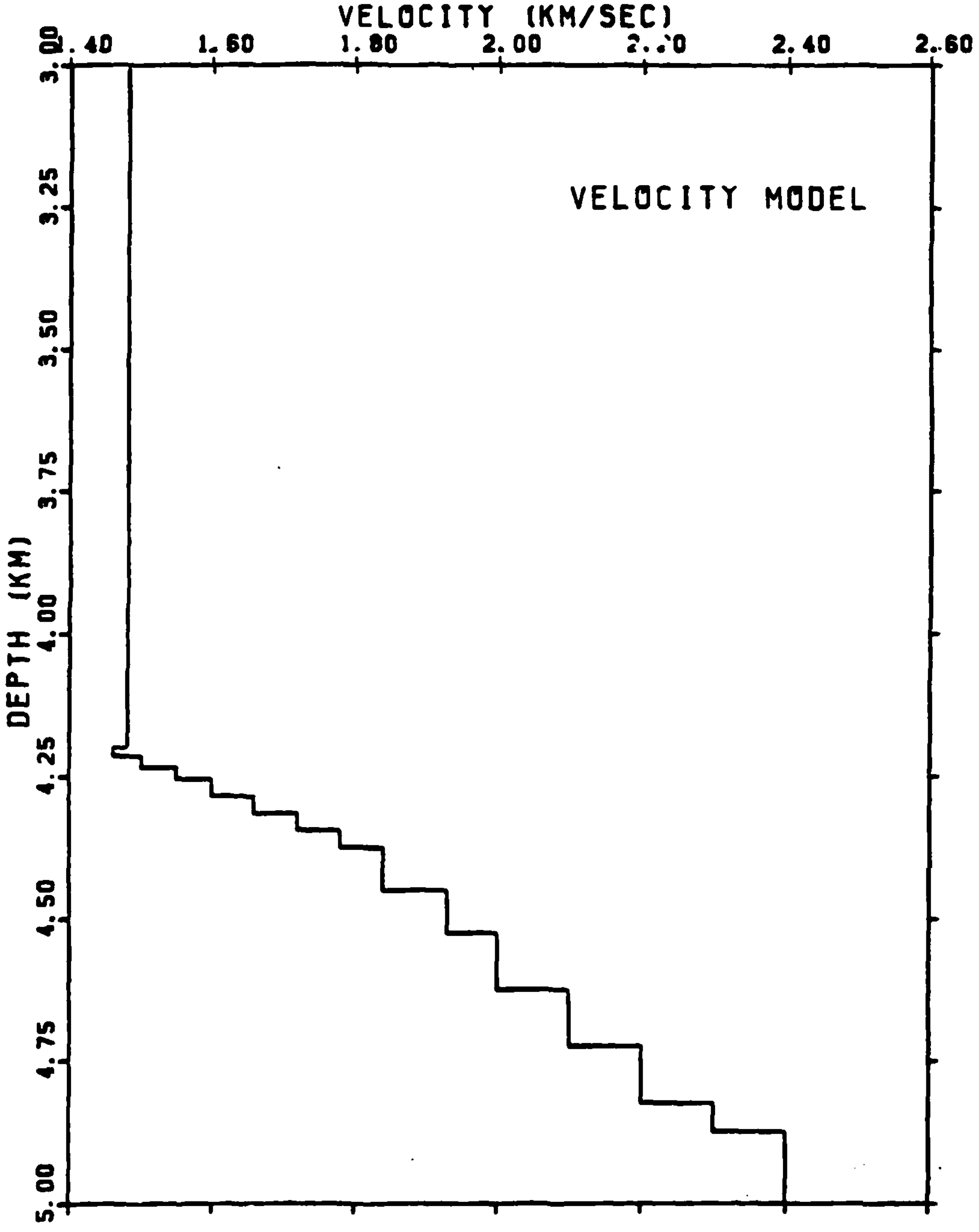

VELOCITY (KM/SEC)
1.40
1.60
1.80
2.00
2.20
2.40
2.60
VELOCITY MODEL
DEPTH (KM)
3.00
3.25
3.50
3.75
4.00
4.25
4.50
4.75
5.00

Fig. 10

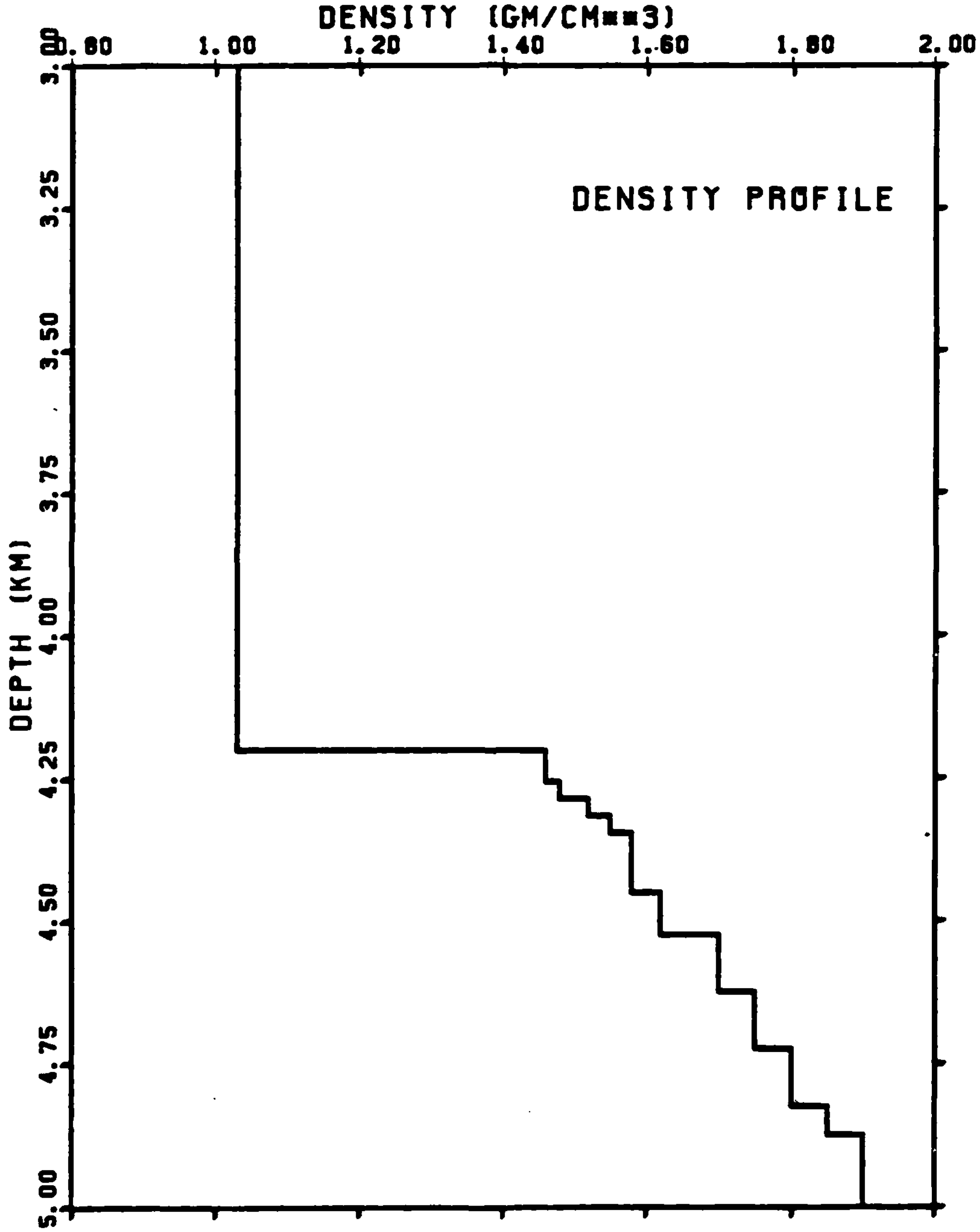

DENSITY (GM/CM**3)
0.80
1.00
1.20
1.40
1.60
1.80
2.00
DENSITY PROFILE
DEPTH (KM)
3.00
3.25
3.50
3.75
4.00
4.25
4.50
4.75
5.00

Fig. 11

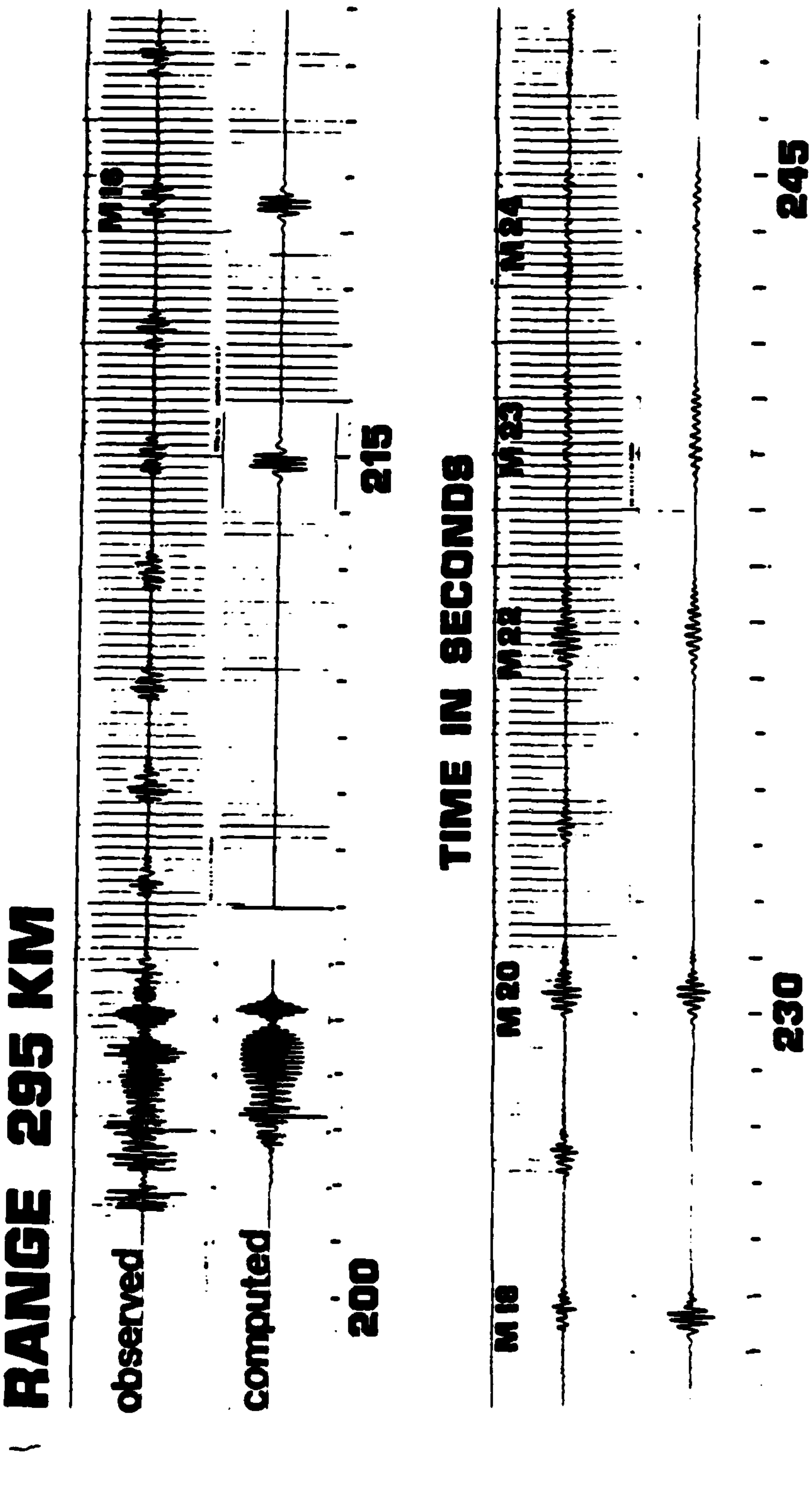
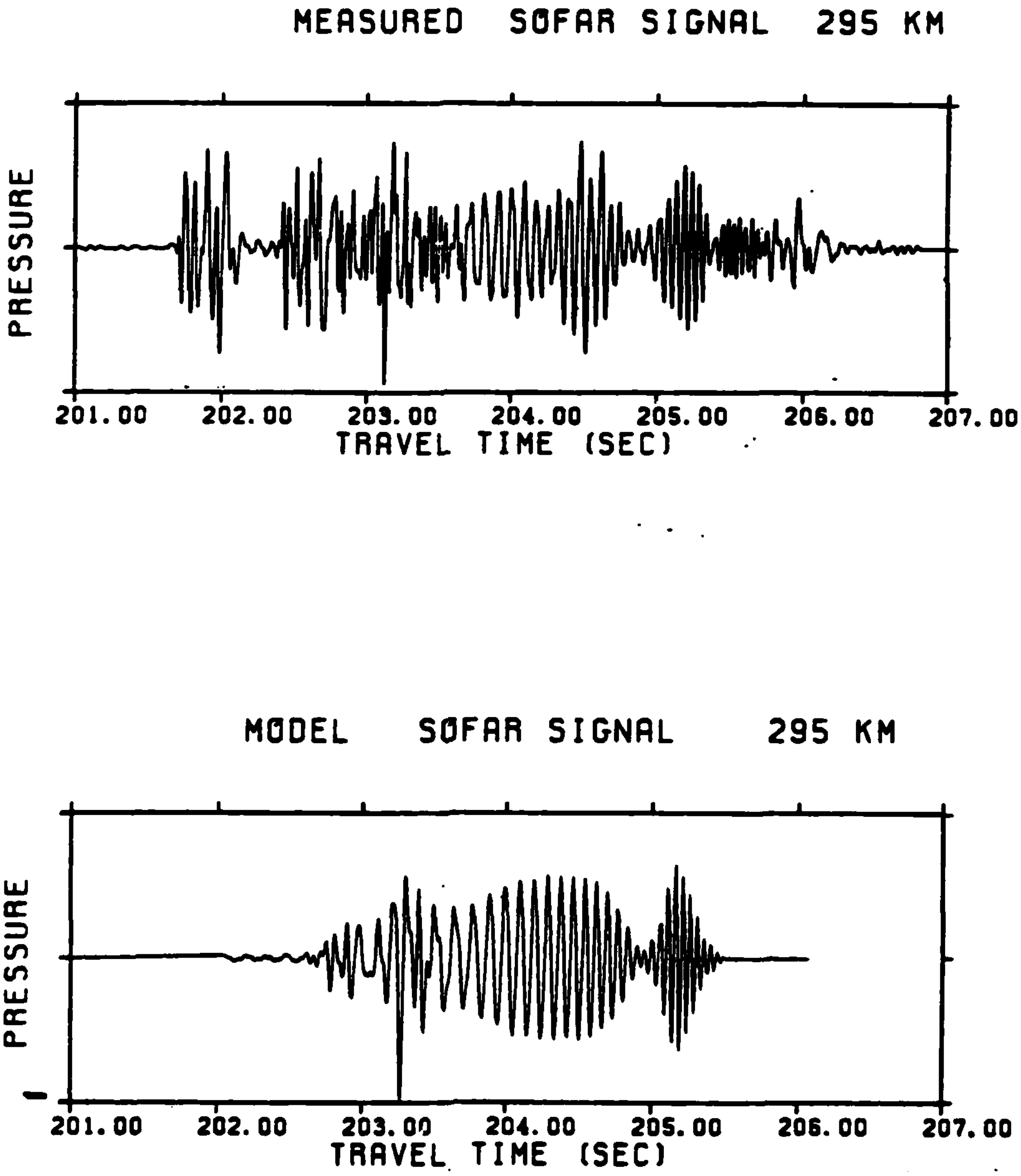

RANGE 295 KM
observed
computed
M16
200
215
TIME IN SECONDS
M18
M20
M22
M23
M24
230
245

Fig. 12



MEASURED SØFAR SIGNAL 295 KM

MØDEL SØFAR SIGNAL 295 KM

Fig. 13

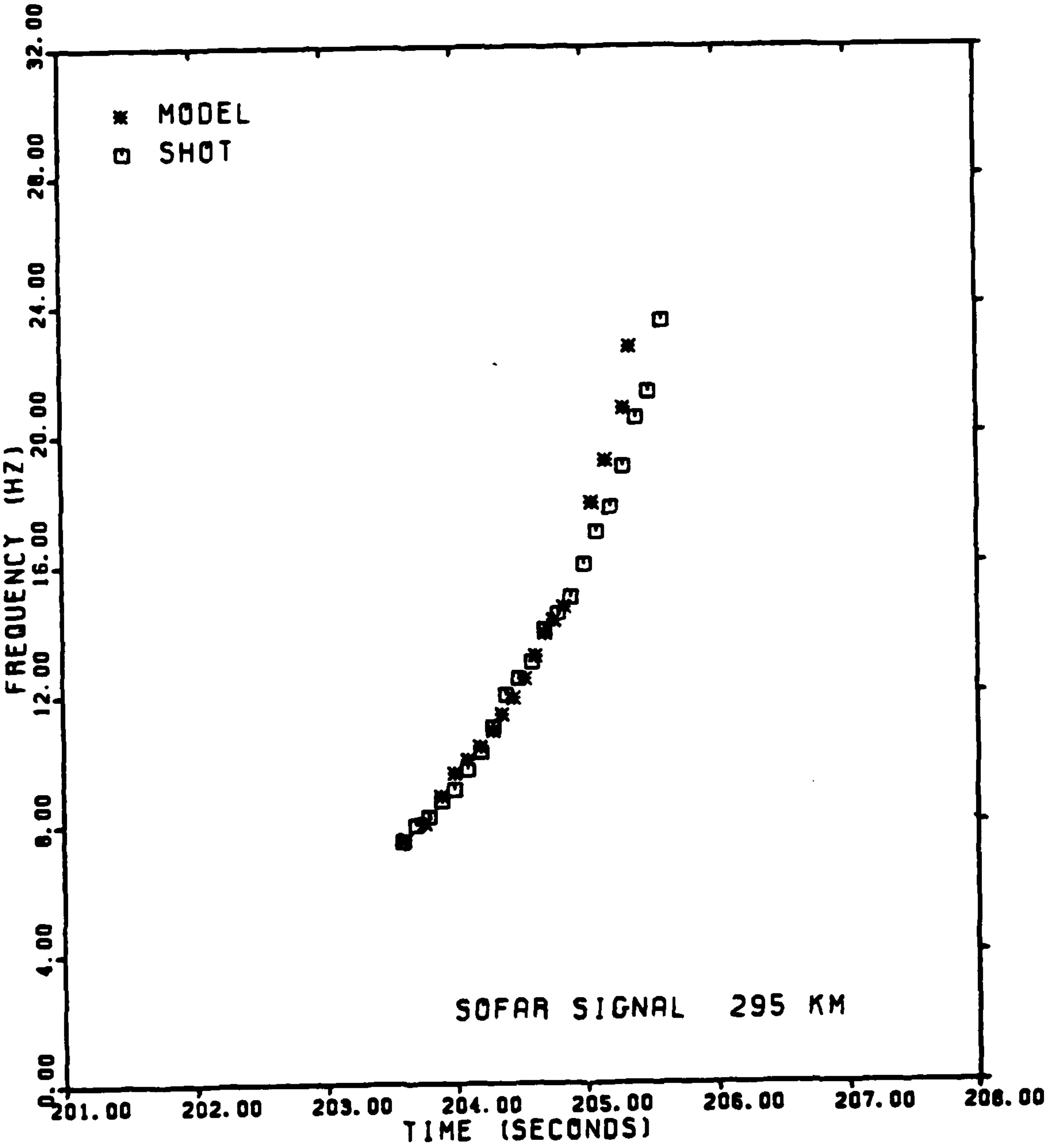

✱ MODEL
□ SHOT
SOFAR SIGNAL 295 KM
FREQUENCY (HZ)
0.00
4.00
8.00
12.00
16.00
20.00
24.00
28.00
32.00
TIME (SECONDS)
201.00
202.00
203.00
204.00
205.00
206.00
207.00
208.00

Fig. 14

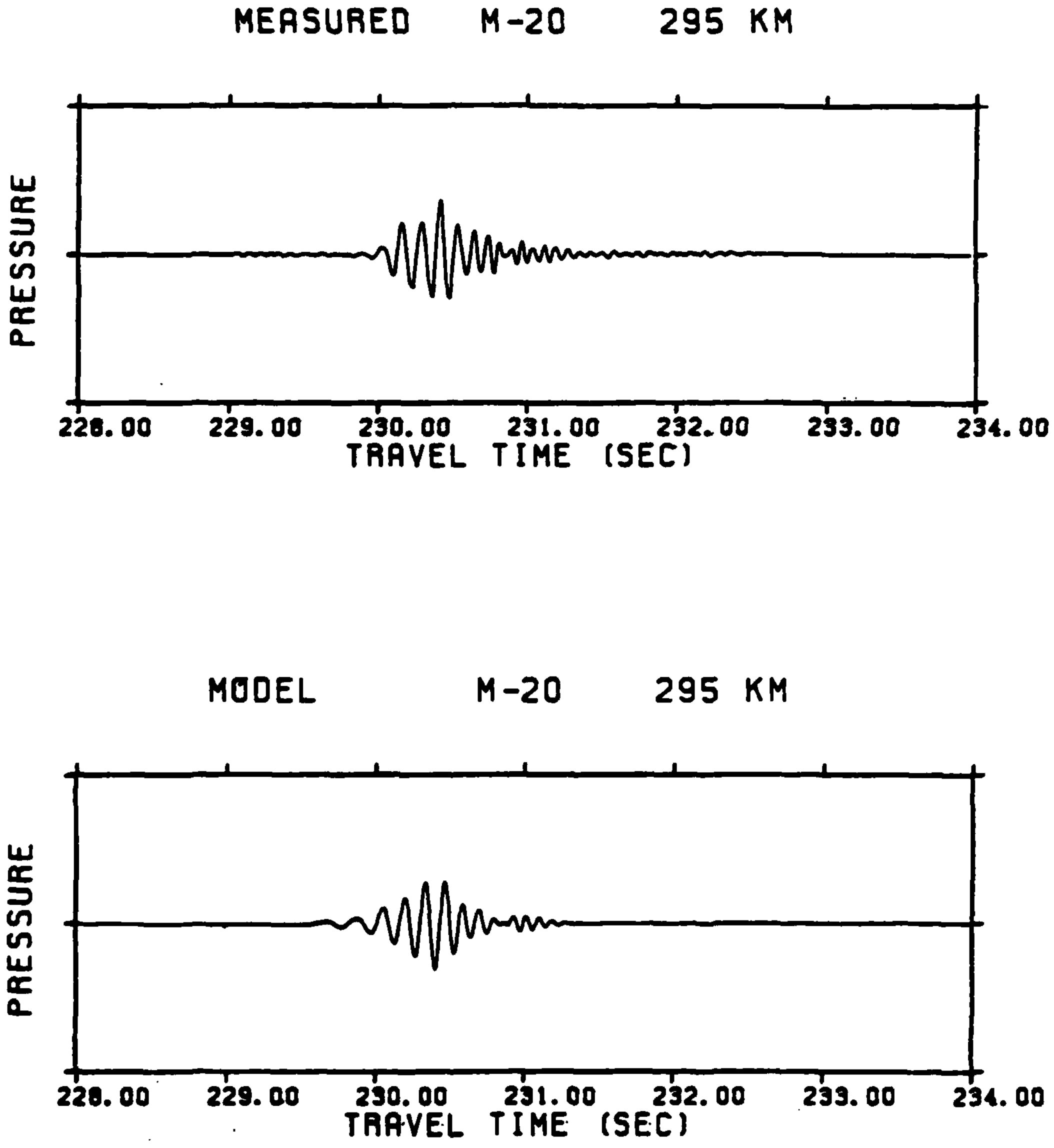

MEASURED M-20 295 KM
PRESSURE
228.00
229.00
230.00
231.00
232.00
233.00
234.00
TRAVEL TIME (SEC)
MODEL M-20 295 KM
PRESSURE
228.00
229.00
230.00
231.00
232.00
233.00
234.00
TRAVEL TIME (SEC)

Fig. 15

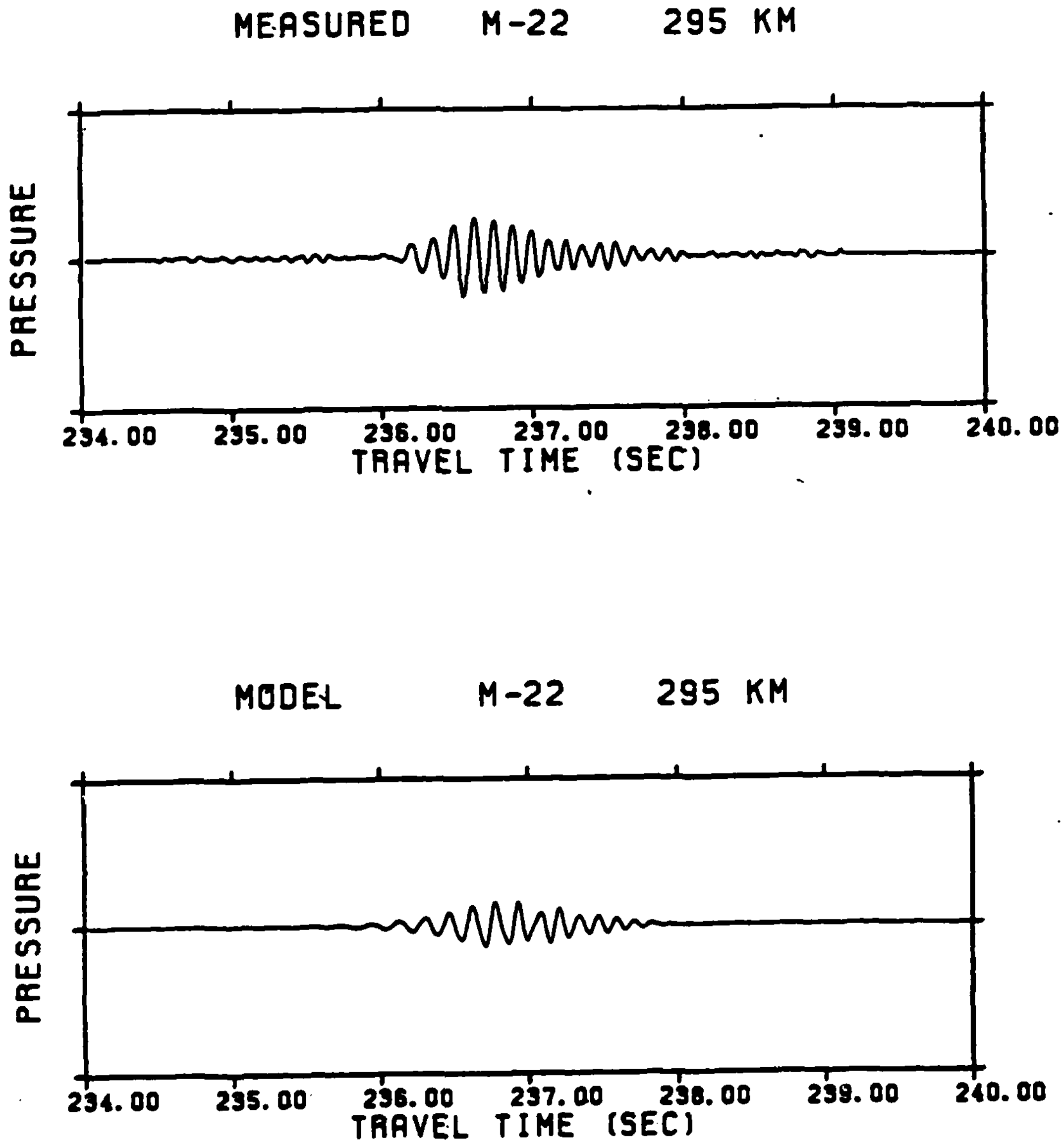
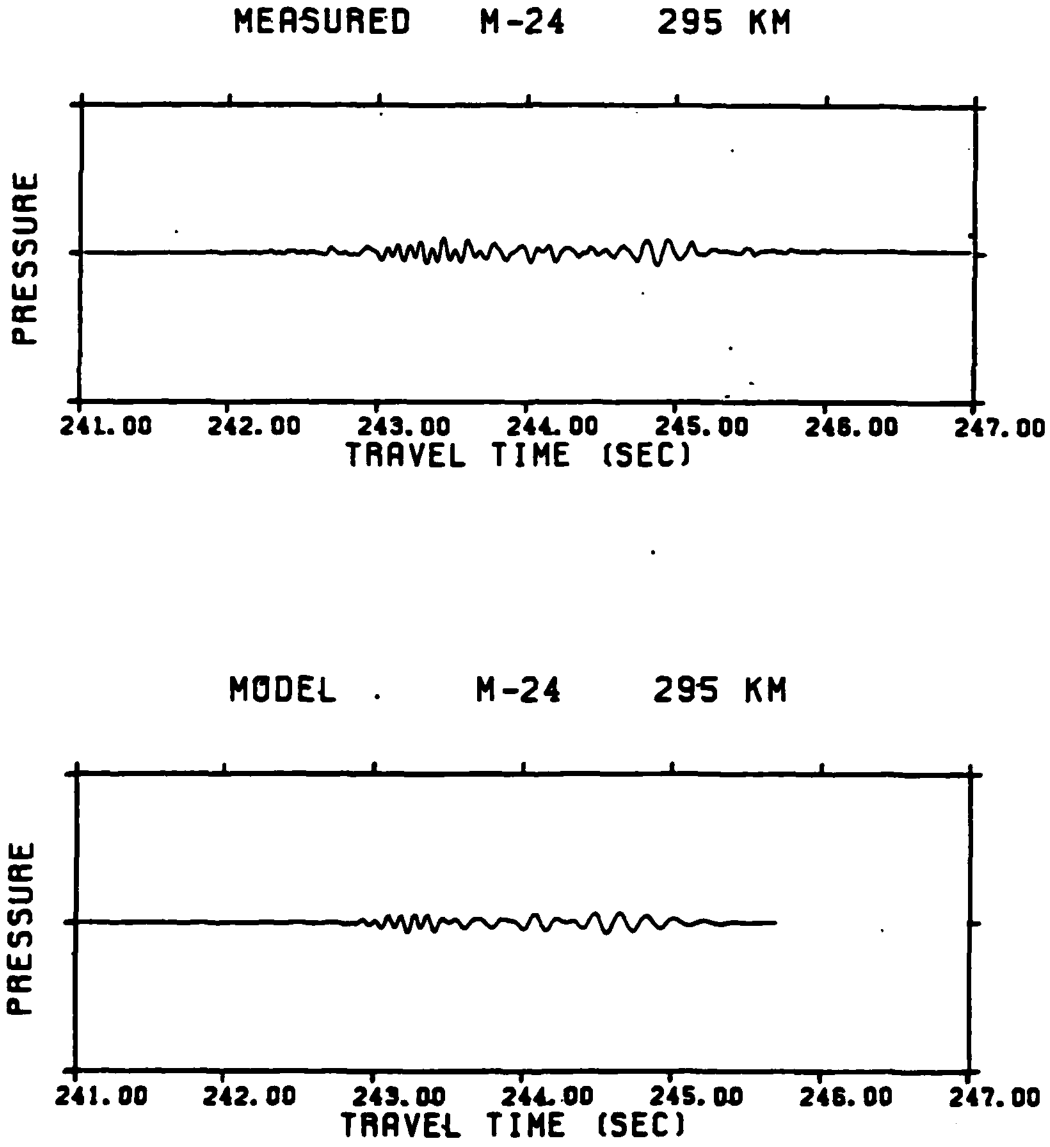

MEASURED M-22 295 KM
PRESSURE
234.00
235.00
236.00
237.00
238.00
239.00
240.00
TRAVEL TIME (SEC)
MODEL M-22 295 KM
PRESSURE
234.00
235.00
236.00
237.00
238.00
239.00
240.00
TRAVEL TIME (SEC)

Fig. 16



MEASURED M-24 295 KM

MODEL M-24 295 KM

Fig. 17

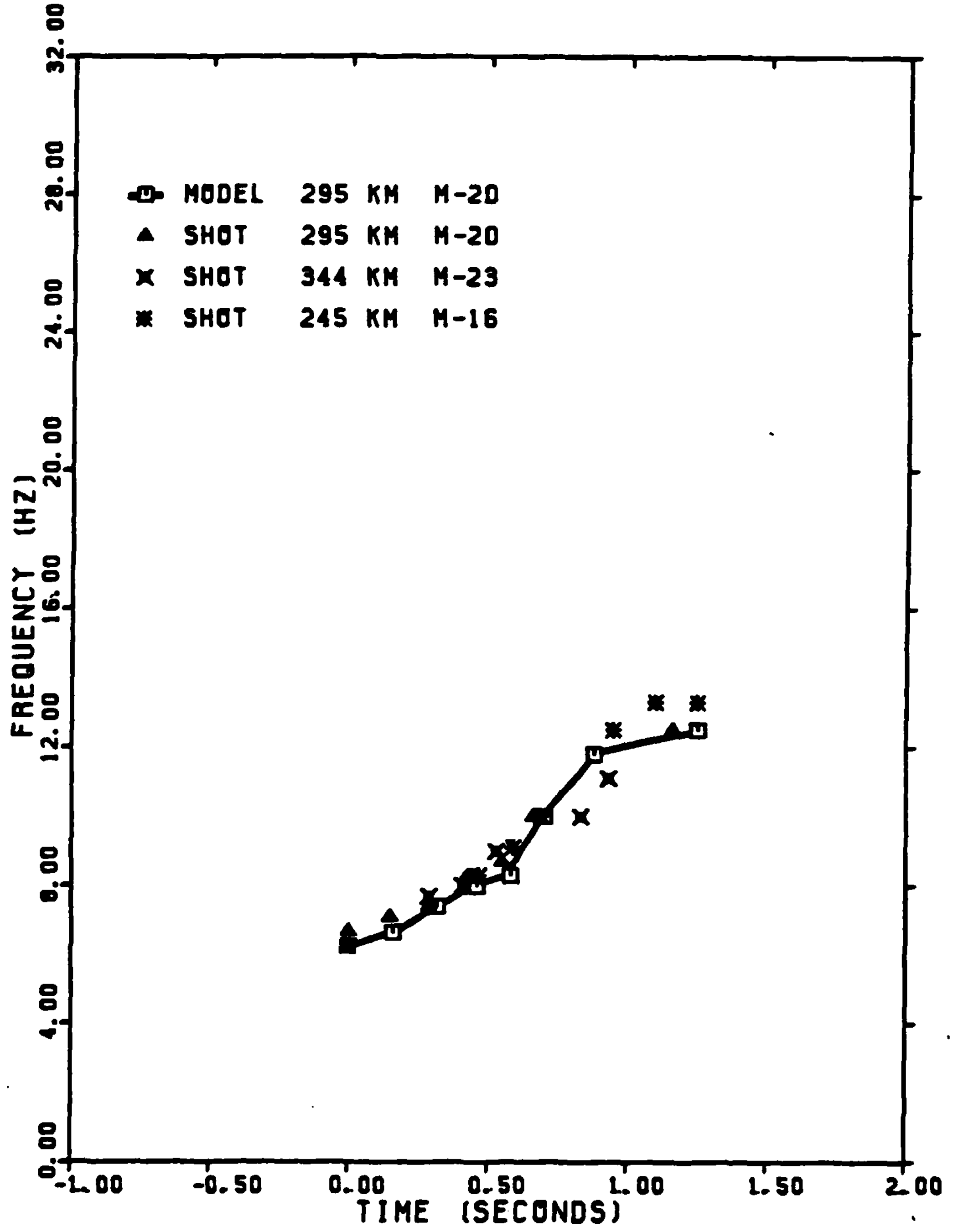

-□- MODEL 295 KM M-20
▲ SHOT 295 KM M-20
✕ SHOT 344 KM M-23
✱ SHOT 245 KM M-16
FREQUENCY (HZ)
0.00
4.00
8.00
12.00
16.00
20.00
24.00
28.00
32.00
-1.00
-0.50
0.00
0.50
1.00
1.50
2.00
TIME (SECONDS)

Fig. 18

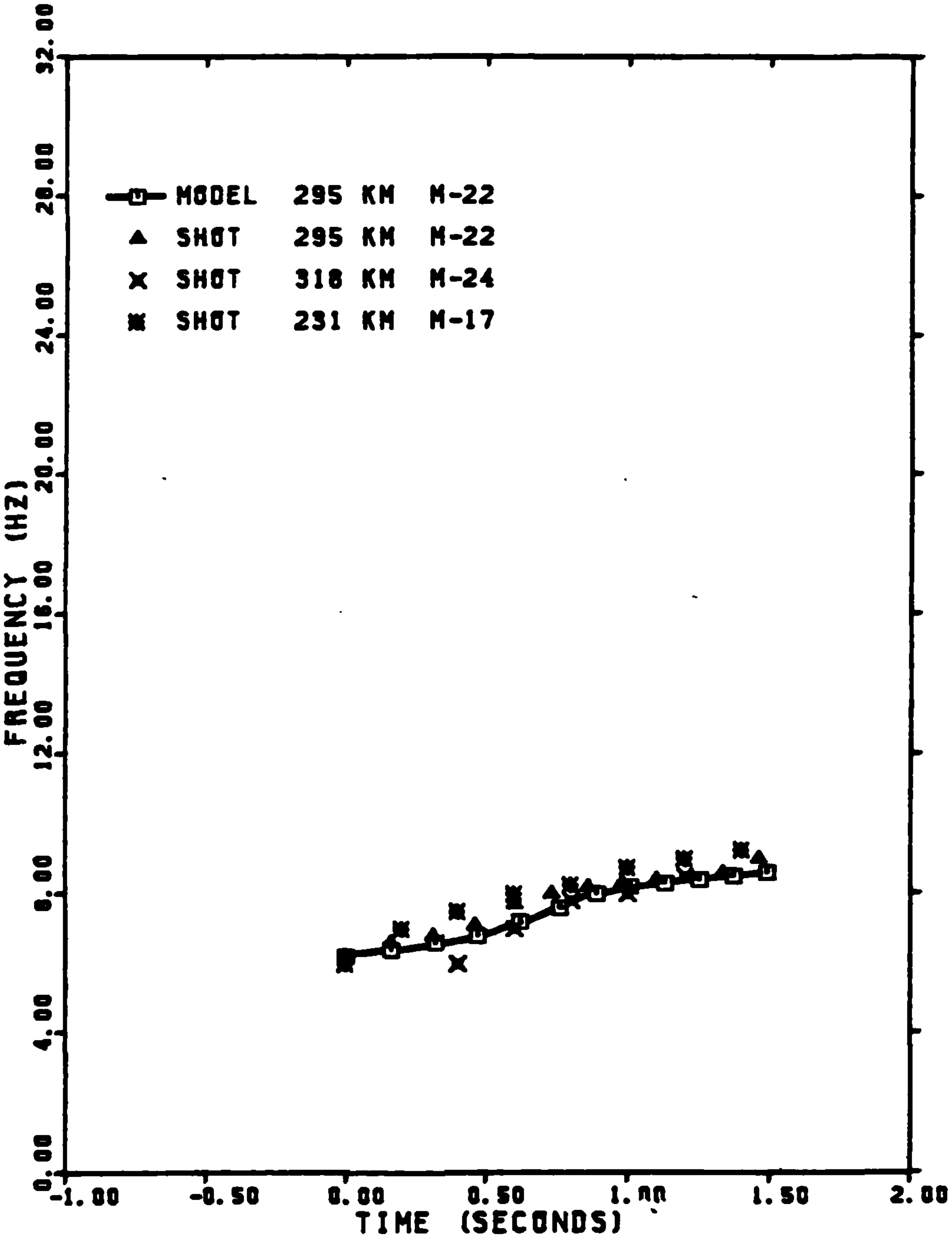
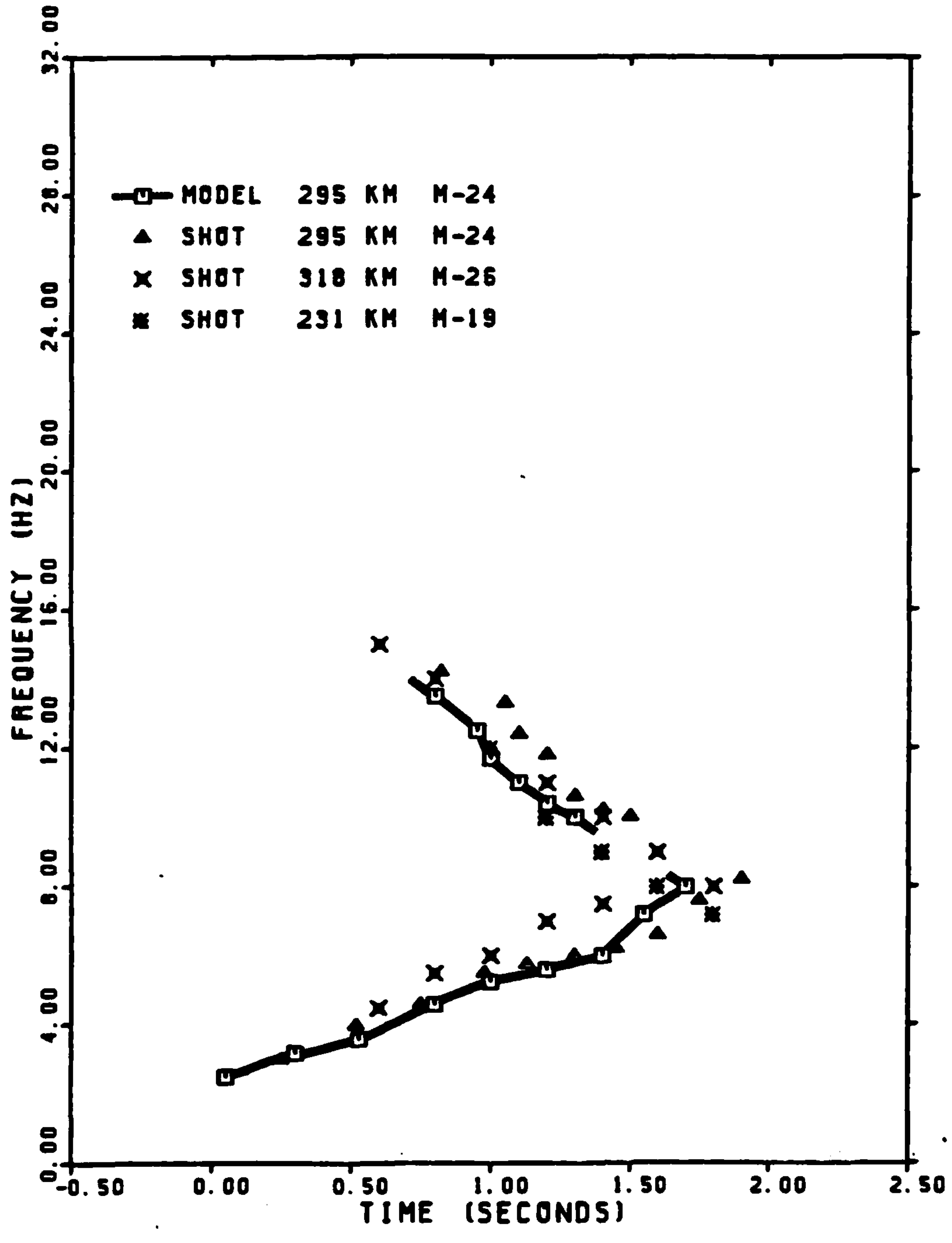

MODEL 295 KM M-22
SHOT 295 KM M-22
SHOT 318 KM M-24
SHOT 231 KM M-17
FREQUENCY (HZ)
0.00
4.00
8.00
12.00
16.00
20.00
24.00
28.00
32.00
TIME (SECONDS)
-1.00
-0.50
0.00
0.50
1.00
1.50
2.00

Fig. 19



—□— MODEL 295 KM M-24
▲ SHOT 295 KM M-24
✕ SHOT 318 KM M-26
✱ SHOT 231 KM M-19
FREQUENCY (HZ)
0.00
4.00
8.00
12.00
16.00
20.00
24.00
28.00
32.00
TIME (SECONDS)
-0.50
0.00
0.50
1.00
1.50
2.00
2.50

Fig. 20

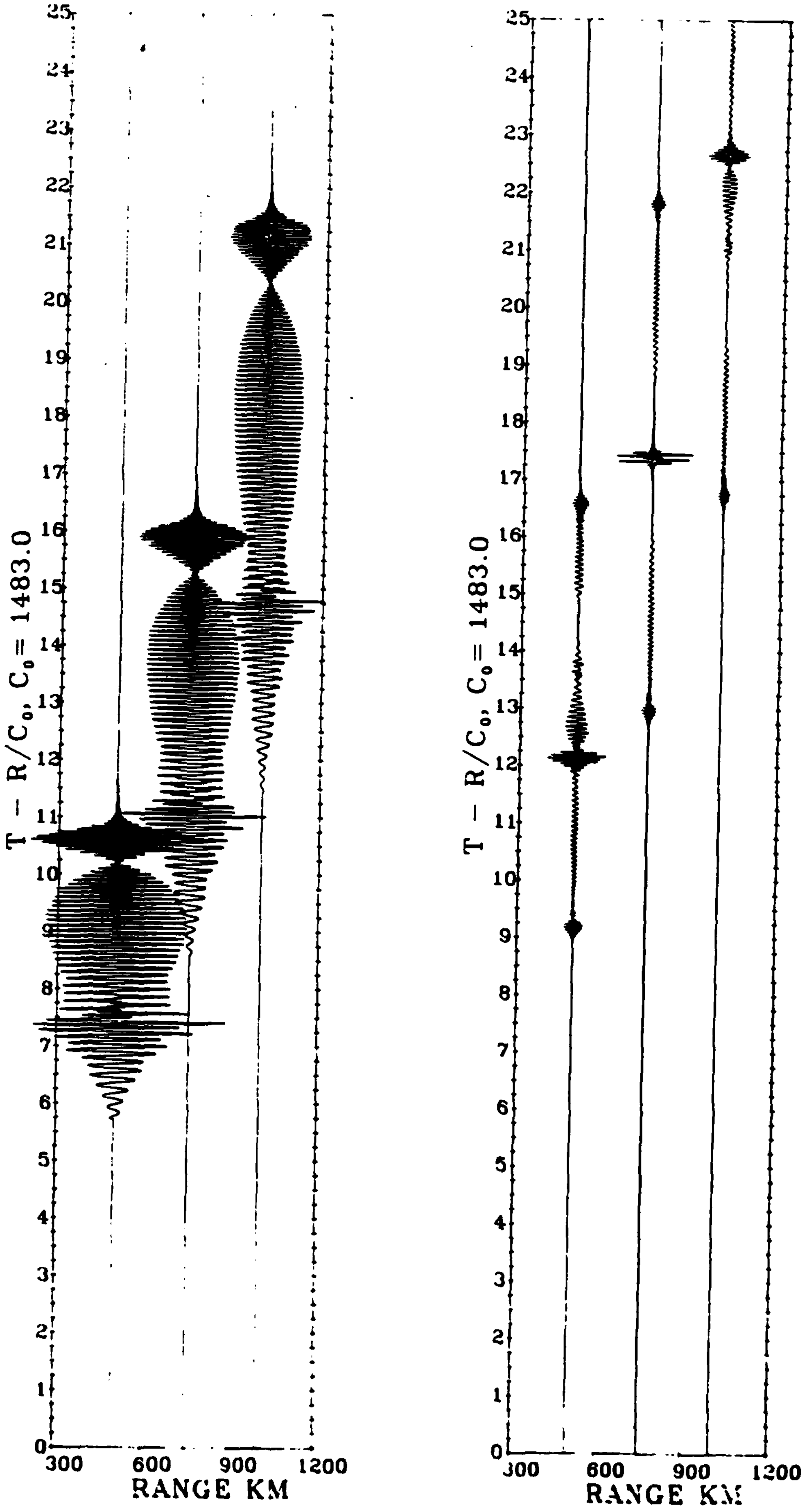

T – R/C₀, C₀= 1483.0
0
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
300
600
900
1200
RANGE KM
T – R/C₀, C₀= 1483.0
0
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
300
600
900
1200
RANGE KM

Fig. 21

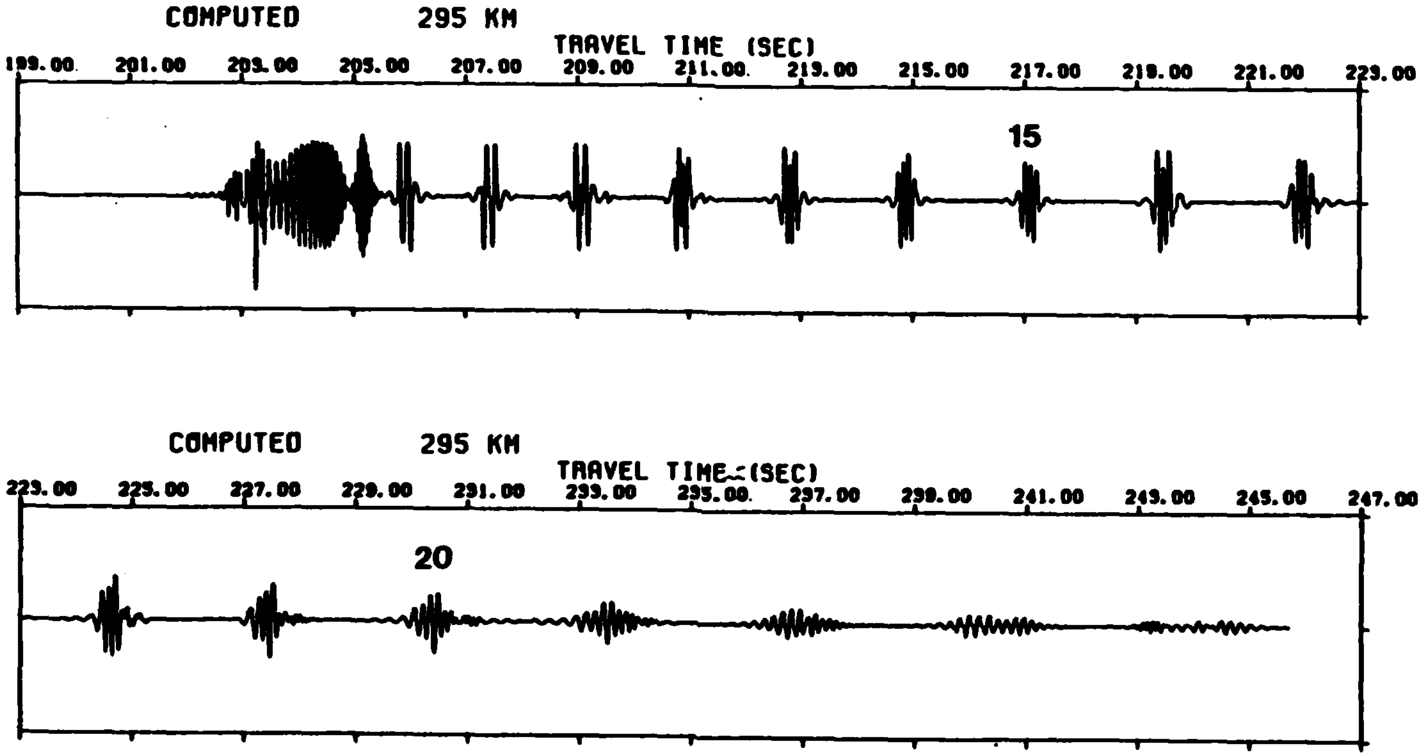

COMPUTED
295 KM
TRAVEL TIME (SEC)
199.00
201.00
203.00
205.00
207.00
209.00
211.00
213.00
215.00
217.00
219.00
221.00
223.00
15
COMPUTED
295 KM
TRAVEL TIME (SEC)
223.00
225.00
227.00
229.00
231.00
233.00
235.00
237.00
239.00
241.00
243.00
245.00
247.00
20

Fig. 22

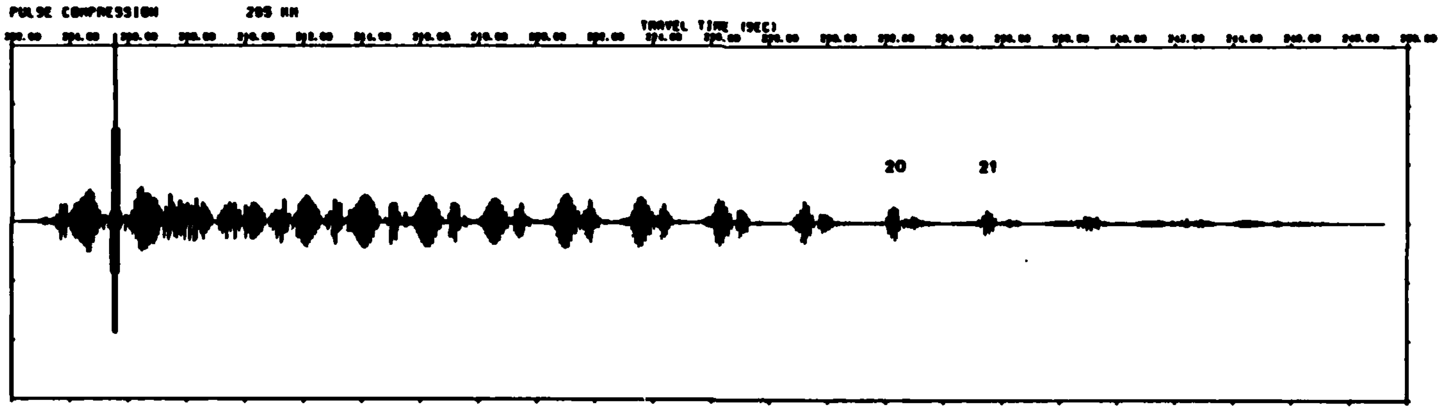


Fig. 23

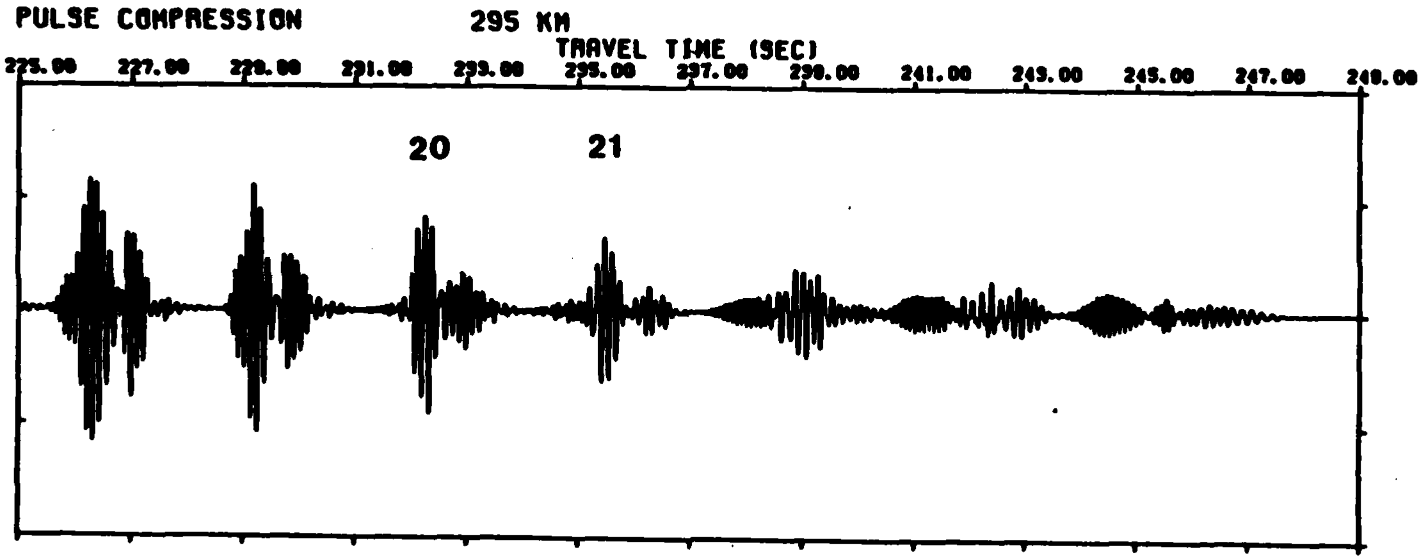

PULSE COMPRESSION
295 KM
TRAVEL TIME (SEC)
225.00
227.00
229.00
231.00
233.00
235.00
237.00
239.00
241.00
243.00
245.00
247.00
249.00
20
21

Fig. 24

Introduction

This volume is being published to present and describe for use the current computer work of Dr. H. Kutschale.

Dr. Kutschale is presently located at the Lamont-Doherty Geological Observatory of Columbia University in Palisades, New York.

The work described here is a continuation of many years of work by Dr. Kutschale in the field of underwater acoustics using the Fast Field Program (FFP). Investigators in the Naval community who have had previous exposure to Dr. Kutschale's work and who have made use of his work will find the current work to be easily incorporated into their own efforts due to its similarity of input with the older versions of these programs. All will find it easily understood due to the step by step style of documentation in each of the user's manuals. In most cases the user groups have received the current programs along with many of the test cases described in this volume.

TLOSS11CC, the first program listed, is the 'basic' program and is an improvement over previous FFP programs due to its ability to handle sound velocity profiles of a more general nature than before. Equal velocity layers are no longer required. Continuously varying velocities are handled using the Airy function. Also, double precision variables enable the program to remain stable when the input is far from the norm, though not to the point of ridiculousness.

TLOSS22CC is a step up with all the features of TLOSS11CC along with the ability to print out the results of the transmission loss due to one particular chosen bottom bounce.

TLOSS10HH is also an enhanced version of TLOSS11CC. This program adds the ability to use the Normal Mode method to all the features of TLOSS11CC.

TLOSS17HH starts with TLOSS10HH as a foundation and adds the ability to calculate the branch integral with the Normal Mode method.

TLOSS51HH gives the user all of the power of TLOSS17HH plus the option of subdividing the input sound velocity profile in order to do step by step integration. Also, calculations of the eigen functions are available using the Normal Mode method and these eigen function values along with the integrand values are put out to disk for further use by other programs.

IMPULS11B is the basic impulse program, corresponding in function to TLOSS11CC.

IMPULS22C is the impulse program that corresponds to TLOSS22CC and has the same BOUNCE equations.

IMPULS11G is the impulse program that corresponds to TLOSS10HH. The Normal Mode method is an option in this program.

In addition to these eight FFP programs Dr. Kutschale is also currently using two Parabolic Equation programs. One program is a transmission loss program, PELOSSX, and one program is an impulse program, PEPULSEX, both of which were developed by Dr. Kutschale in conjunction with Dr. F. Tappert.

These ten programs are put forth here TO ENABLE A PERSON TO USE THESE PROGRAMS, and for no other reason. One must delve deeper than this volume in order to revise any of these programs to meet a special need not as yet addressed. What is included for each program is:

1. A User's Manual describing input and special features
2. The input for several selected test cases
3. The output for these same selected test cases

TLOSS11CC USER'S MANUAL

A. Introduction

TLOSS11CC, our streamlined transmission loss (hereafter abbreviated as TL) program, was completed in October of 1982.

TLOSS11CC has these features and options for the user:

1. The FFP computational method

3. A comprehensive selection of modelling parameters. These include:
   a. A detailed sound velocity profile
   b. Choice of source depth
   c. Choice of detector depth
   d. Choice of frequency
   e. Choice of wave number integrand limits
   f. Optional beaming, as it is affected by integrand tapering (in the FFP mode)
   g. Optional surface roughness
   h. Choice of either fluid or vacuum upper half space
   i. Possibility of the source and detector in the fluid upper half space
   j. Optional internal roughness

4. Output. This includes:
   a. A printout of all input
   b. The TL plot
   c. The wave integrand plot of the FFP; either short or long.

   Note: All short plots except the wave integrands are repeated, the repeated plot being a smoothed copy of the first and having the same bounds as the first.

5. Additional Optional Tables and Plots. These include:
   a. The TL plot in nautical miles, if desired and if the range is sufficiently long. The original TL plot is in kilometers.

6. Numerical output of the TL for further processing or plotting. Formerly, this was called punched output, since it went out to the card punch.

B. Input Variables and Formats

Program input for TLOSS11CC includes both the physical model and program control. This section of the manual will list and explain each line of the input.

1. Entry 1, one line

MMAXF, NCARD, NREAD — variable names
(2I3) — format

MMAXF is a control variable which is used to read in the correct number of points in the sound velocity profile. (see 2. - next Entry)

NCARD is a control variable that, when set non-zero allows the program to put out TL data on unit 7 in a form which is readable as input by another program. A zero value will prevent any output on unit 7. This is normally used to make graphic plots. (Originally, on the IBM 360/91, unit 7 was the punched card unit, hence the name NCARD.)

NREAD is a control variable that, when set non-zero, directs the program to interpret Entry 2 data (see below) as thickness input instead of the usual depth input.

2. Entry 2, MMAXF lines (max. of 49 lines)

((V(M,N), N=1,7), M=1,MMAXF) — variable name
(7D10.0) — format

V is an array that contains the sound velocity profile. For each point in the profile, and on each line, (one point per line), the program expects:

a. a depth value, in meters, except when NREAD is set non-zero, in which case the program expects a thickness value, in meters
b. a sound (pressure) velocity value, in meters per second
c. a sound (shear) velocity value, in meters per second, if applicable
d. a corresponding attenuation value for pressure
e. a corresponding attenuation value for shear, if applicable
f. a density value, in grams per c.c., or specific density
g. a roughness value, if applicable

Note: No depth value can be less than a previous depth value. The program will stop if this is so.

3. Entry 3, one line

KSF, LRF, MM, LS, KKK, IBEAM, NW, FREQ, PO, C, CMAX — variable names
(7I5, 4D10.0) — format

KSF is a pointer to the source depth value. It points to one of the points in the sound velocity profile; the one which is the source depth value.

LRF is a pointer to the detector (or receiver) depth value. It's function is similar to KSF.

MM is a TL plot pruning factor, with a minimum acceptable value of 2. MM is applicable to FFP only. If an MM value of N is read in, then the first 1/N of the plot will be plotted. This is useful either to get a more detailed look at the output or to prune away that part of the plot affected by range aliasing.

LS is the number of points used in the wave integration. It also affects the range of the TL plot, for the FFP method.

KKK is a control variable that indicates:
a. If zero, this is the last case.
b. If negative, the next case starts with Entry 1, (new profile).
c. If positive, the next case starts with Entry 3.

IBEAM is a control variable that, when set non-zero, requires the program to taper the wave integrand at each end.

NW is a flag, that when set non-zero, directs the program to put out a printer plot of the wave integrand.

FREQ is the frequency for this case, in Hertz.

PO is the source level. For TL PO = 1.0

C is the lower bound sound velocity in the wave integration, in meters per second.

CMAX is the upper bound sound velocity in the wave integration, in meters per second. If a zero is entered here, the wave number integration will start from zero.

4. Entry 4, one line

```
REFLTO, ALPHAO, ATTENO, RHOO, H, INAUT, LONG          variable name
( 5D10.0, 2I5 )                                       format
```

REFLTO is the RMS surface roughness value, in meters. If this value is non-zero, the program explicitly sets the next three values to zero.

ALPHAO is, if present, the sound velocity in the fluid upper half space, in meters per second. This is the variable that determines whether the present case has either a fluid or vacuum upper half space.

ATTENO is, if present, the attenuation in the fluid upper half space.

RHOO is, if present, the density of the fluid upper half space, in grams per c.c.

H is, if present, the distance between the source and detector in the fluid upper half space. To use this option:

a. The first two sound velocity depths must be zero.
b. There must be some variation in the pressure sound velocity profile.
c. There must not be any shear sound velocity.
d. ALPHAO and H must be present.

INAUT is a control variable that, when set non-zero, allows the plotting of the TL plot in nautical miles if the range is greater than 100 kilometers.

LONG is a control variable that, when set non-zero, switches the wave integrand plot from short to long. This is used when it is necessary to examine each point of the wave integrand.

C. Operating Procedures

The primary computer at Lamont-Doherty Geological Observatory is a VAX 11/780, which runs under the VMS operating system. Listed below are four distinctly different operations that are used to process TLOSS11CC.

1. Compiling-Linking

a. Compiling (FORT command)

Compiling the TLOSS11CC program on the VAX 11/780 has always been done with the following switches on:

1. /CHECK Forbids array pointer overflow.
2. /NOI4 Integer variables are set to 2 bytes.
3. /NOF77 Where the compiler needs to choose between Fortran 77 and Fortran 4+, the latter is chosen.
4. /LIST A compiler listing is made and stored on the disk as TLOSS11CC.LIS.

b. Linking (LINK command)

Linking is straight-forward. No extra routines are linked with TLOSS11CC.

2. Running

TLOSS11CC is normally run through a batch stream, in particular the SLOW batch stream, since this is the cheapest route, especially at night. However, there is another batch stream with higher priority (and higher cost) that can be used for quicker results, the SYS$BATCH batch stream. In order to use either of these two batch streams:

a. The executable file (TLOSS11CC.EXE) must be ready to use. The LINK operation satisfies this requirement.

b. The data file must also be ready to use. This is normally prepared by the user using the editor, (and the TLOSS11CC USER'S MANUAL, if necessary) at a terminal.

c. Submit the command file, TLOSS11CC.COM, to the batch stream (or batch queue) using a command such as one of these:

```
SUBMIT/NAME=LEC01OCT/QUEUE=SLOW/AFTER=20:00 TLOSS11CC
       SLOW queue, Job starts at 8:00 P.M.

SUBMIT/NAME=LEC01OCT/QUEUE=SLOW TLOSS11CC
       SLOW queue, Job starts now

SUBMIT/NAME=LEC01OCT TLOSS11CC
       SYS$BATCH queue ( the 'regular' queue ),
       Job starts now
```

d. The following command file, also prepared using the editor, is a typical file submitted, using one of the previous submit commands.

```
$ASSIGN LEC01OCT SYS$PRINT                      (1)
$ASSIGN LEC01OCT SYS$OUTPUT                     (2)
$ASSIGN DRO:LEC01OCT.PLT FOR007                 (3)
$ASSIGN DRO:LEC01OCT.OUT FOR006                 (4)
$ASSIGN DRO:STANDARD.DAE FOR005                 (5)
$TYPE TLOSS11CC                                 (6)
$COPY DRO:STANDARD.DAE DRO:LEC01OCT.DAT         (7)
$RUN TLOSS11CC                                  (8)
```

This run is called the LEC01OCT run. 'LEC' stands for LOSS ELEVEN C. '01OCT' indicates that this is the first run in October.

The first two lines in this command file put the log file on the disk instead of the printer.
The third line puts unit 7 output into file LECO1OCT.PLT.
The fourth line puts unit 6 output into file LECO1OCT.OUT.
The fifth line draws unit 5 input from STANDARD.DAE.
The sixth line copies TLOSS11CC.LIS to LECO1OCT.LIS.
The seventh line copies STANDARD.DAE to LECO1OCT.DAT.
The eighth line runs program TLOSS11CC.

A command file such as this yields a file group; each file having the same name but a different extension. These files are then treated collectively. They are printed together and stored on tape together. Such a group is, for example:

LECO1OCT.DAT	a copy of the data file
LECO1OCT.LIS	the program listing file
LECO1OCT.LOG	the log file
LECO1OCT.OUT	the output file
LECO1OCT.PLT	the numerical output file

The user normally gets a complete printout of the run by typing the command:

```
PRINT LECO1OCT.*;*
```

on the terminal.

3. Copying Runs to Tape (Archiving)

Periodically, it is necessary to clear the old runs off the disk and put them on tape. This is a very simple operation and proceeds as follows:

a. Initialize a tape. Give it a name that is somehow indicative of the present month. For example:

```
INITIALIZE/DENSITY=1600 MT1: OCTNOV
```

This tape would have runs made in October and November. This step is not necessary if the user is adding runs to a tape that is already partly filled. In such a case, be careful not to initialize!

b. Mount the tape with the MOUNT command.

```
MOUNT MT1: OCTNOV
```

This is not the same as physically mounting the tape on the tape drive, which is assumed to have taken place previously.

c. Copy all the run files from disk to tape with the COPY command. For example in this case:

```
COPY *OCT.*;* MT1:
COPY *NOV.*;* MT1:
```

would copy all the October and all the November runs onto the tape mounted on tape drive 1.

d. For future reference, it is good to get a list of names of the files on the tape. This is accomplished as follows:

```
DIRECTORY/PRINT MT1:
```

The printout resulting from this command does not have the tape name on it, so the user should write this on immediately.
If, for some reason, this printout is lost, the tape can be remounted and the directory can be taken again. But this is no reason to be careless with the original directory printout.

e. Dismount the tape with the dismount command:

```
DISMOUNT MT1:
```

f. Delete the files on the disk that have just been copied to the tape. In this case:

```
DELETE *OCT.*;*
DELETE *NOV.*;*
```

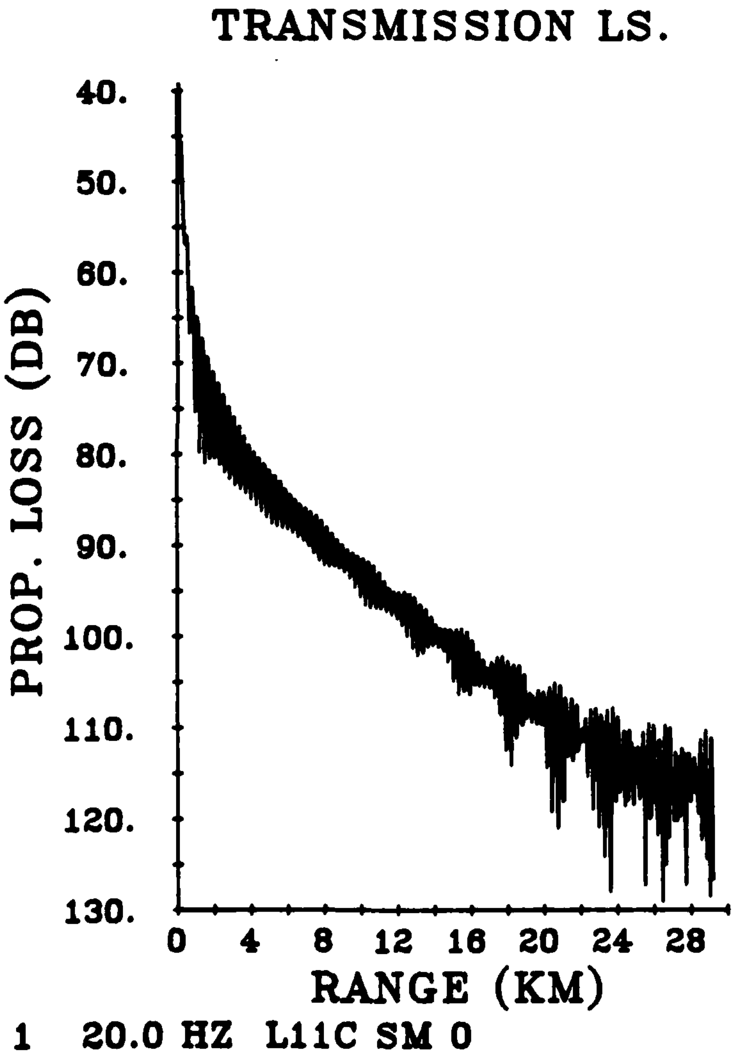
4. Distributing the Program by Tape

The program and test cases are distributed using tapes with the same format as the archival tapes. This is the Files-11 format, peculiar to Digital Equipment Corporation. The tapes from this system have a density of 1600 bpi, and a blocksize of 2048 bytes.

If any further questions arise, feel free to call:

Dr. H. Kutschale	(914) 359-2900 x382
Mr. L. Carroll	(914) 359-2900 x221

```
LECO1OCT.DAT;2                                  6-APR-1983 15:30
                         TEST CASE DATA
 06  1
  0.00      1500.0                    .0000150     DATA FOR TLOSS11CC
 50.00      1500.0                    .0000150                1.00
 100.0      1500.0                    .0000150                1.00
 100.0      3000.0      1400.0        .00007700               1.00
 500.0      3000.0      1400.0        .00007700   .0001600 2.50
 500.0      3000.0      1400.0        .00007700   .0001600 2.50
    2     3     02 1024   00   000    1   020.0   .0001600 2.50
  0.0        1500.0   .0000150   1.00       0000.0   10000.      1000.0     8000.0
 00.          00.0                                          0     1    00     1
```


TRANSMISSION LS.
PROP. LOSS (DB)
40.
50.
60.
70.
80.
90.
100.
110.
120.
130.
0
4
8
12
16
20
24
28
RANGE (KM)
1 20.0 HZ L11C SM 0

TLOSS22CC USER'S MANUAL

A. Introduction

TLOSS22CC, one of our streamlined transmission loss (hereafter abbreviated as TL) programs, was completed in March of 1983.

TLOSS22CC has these features and options for the user:

1. The FFP computational method
   a. The complete, unabridged version
   b. An abbreviated version to calculate the effect of one particular number of bounces

3. A comprehensive selection of modelling parameters. These include:
   a. A detailed sound velocity profile
   b. Choice of source depth
   c. Choice of detector depth
   d. Choice of frequency
   e. Choice of wave number integrand limits
   f. Optional beaming, as it is affected by integrand tapering (in the FFP mode)
   g. Optional surface roughness
   h. Choice of either fluid or vacuum upper half space
   i. Possibility of the source and detector in the fluid upper half space
   j. Optional internal roughness

4. Output. This includes:
   a. A printout of all input
   b. The TL plot
   c. The wave integrand plot of the FFP; either short or long.

   Note: All short plots except the wave integrands are repeated, the repeated plot being a smoothed copy of the first and having the same bounds as the first.

5. Additional Optional Tables and Plots. These include:
   a. The TL plot in nautical miles, if desired and if the range is sufficiently long. The original TL plot is in kilometers.

6. Numerical output of the TL for further processing or plotting. Formerly, this was called punched output, since it went out to the card punch.

B. Input Variables and Formats

Program input for TLOSS22CC includes both the physical model and program control. This section of the manual will list and explain each line of the input.

1. Entry 1, one line
MMAXF, NCARD, NREAD — variable names
(2I3) — format

MMAXF is a control variable which is used to read in the correct number of points in the sound velocity profile. (see 2. - next Entry)

NCARD is a control variable that, when set non-zero allows the program to put out TL data on unit 7 in a form which is readable as input by another program. A zero value will prevent any output on unit 7. This is normally used to make graphic plots. (Originally, on the IBM 360/91, unit 7 was the punched card unit, hence the name NCARD.)

NREAD is a control variable that, when set non-zero, directs the program to interpret Entry 2 data (see below) as thickness input instead of the usual depth input.

2. Entry 2, MMAXF lines (max. of 49 lines)
((V(M,N), N=1,7), M=1,MMAXF) — variable name
(7D10.0) — format

V is an array that contains the sound velocity profile. For each point in the profile, and on each line, (one point per line), the program expects:

a. a depth value, in meters, except when NREAD is set non-zero, in which case the program expects a thickness value, in meters
b. a sound (pressure) velocity value, in meters per second
c. a sound (shear) velocity value, in meters per second, if applicable
d. a corresponding attenuation value for pressure
e. a corresponding attenuation value for shear, if applicable
f. a density value, in grams per c.c., or specific density
g. a roughness value, if applicable

Note: No depth value can be less than a previous depth value. The program will stop if this is so.

3. Entry 3, one line
KSF, LRF, MM, LS, KKK, IBEAM, NW, FREQ, PO, C, CMAX — variable names
(7I5, 4D10.0) — format

KSF is a pointer to the source depth value. It points to one of the points in the sound velocity profile; the one which is the source depth value.

LRF is a pointer to the detector (or receiver) depth value. It's function is similar to KSF.

MM is a TL plot pruning factor, with a minimum acceptable value of 2. MM is applicable to FFP only. If an MM value of N is read in, then the first 1/N of the plot will be plotted. This is useful either to get a more detailed look at the output or to prune away that part of the plot affected by range aliasing.

LS is the number of points used in the wave integration. It also affects the range of the TL plot, for the FFP method.

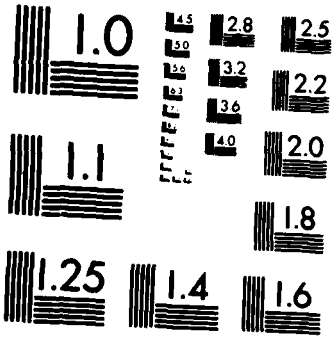
KKK is a control variable that indicates:

a. If zero, this is the last case.
b. If negative, the next case starts with Entry 1, (new profile).
c. If positive, the next case starts with Entry 3.

IBEAM is a control variable that, when set non-zero, requires the program to taper the wave integrand at each end.

AD-A147 492 ARCTIC MARINE ACOUSTICS(U) LAMONT-DOHERTY GEOLOGICAL OBSERVATORY PALISADES NY H W KUTSCHALE OCT 84 LDGO05-20081 N00014-80-C-0021 2/3

UNCLASSIFIED F/G 20/1 NL



MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS-1963-A

NW is a flag, that when set non-zero, directs the program to put out a printer plot of the wave integrand.

FREQ is the frequency for this case, in Hertz.

PO is the source level. For TL PO = 1.0

C is the lower bound sound velocity in the wave integration, in meters per second.

CMAX is the upper bound sound velocity in the wave integration, in meters per second. If a zero is entered here, the wave number integration will start from zero.

4. Entry 4, one line
REFLTO, ALPHAO, ATTENO, RHOO, H, INAUT, LONG, BOUNCE variable name
(5D10.0, 3I5) format

REFLTO is the RMS surface roughness value, in meters. If this value is non-zero, the program explicitly sets the next three values to zero.

ALPHAO is, if present, the sound velocity in the fluid upper half space, in meters per second. This is the variable that determines whether the present case has either a fluid or vacuum upper half space.

ATTENO is, if present, the attenuation in the fluid upper half space.

RHOO is, if present, the density of the fluid upper half space, in grams per c.c.

H is, if present, the distance between the source and detector in the fluid upper half space. To use this option:

a. The first two sound velocity depths must be zero.
b. There must be some variation in the pressure sound velocity profile.
c. There must not be any shear sound velocity.
d. ALPHAO and H must be present.

INAUT is a control variable that, when set non-zero, allows the plotting of the TL plot in nautical miles if the range is greater than 100 kilometers.

LONG is a control variable that, when set non-zero, switches the wave integrand plot from short to long. This is used when it is necessary to examine each point of the wave integrand.

BOUNCE is a control variable that, when set non-zero, directs the program to calculate only the effects of a certain number of bounces, BOUNCE being that number. This is somewhat similar to the 'H variable' output and should not be used in conjunction with that feature.

C. Operating Procedures

The primary computer at Lamont-Doherty Geological Observatory is a VAX 11/780, which runs under the VMS operating system. Listed below are four distinctly different operations that are used to process TLOSS22CC.

1. Compiling-Linking

a. Compiling (FORT command)
Compiling the TLOSS22CC program on the VAX 11/780 has always been done with the following switches on:

1. /CHECK Forbids array pointer overflow.
2. /NOI4 Integer variables are set to 2 bytes.
3. /NOF77 Where the compiler needs to choose between Fortran 77 and Fortran 4+, the latter is chosen.
4. /LIST A compiler listing is made and stored on the disk as TLOSS22CC.LIS.

b. Linking (LINK command)
Linking is straight-forward. No extra routines are linked with TLOSS22CC.

2. Running

TLOSS22CC is normally run through a batch stream, in particular the SLOW batch stream, since this is the cheapest route, especially at night. However, there is another batch stream with higher priority (and higher cost) that can be used for quicker results, the SYS$BATCH batch stream. In order to use either of these two batch streams:

a. The executable file (TLOSS22CC.EXE) must be ready to use. The LINK operation satisfies this requirement.

b. The data file must also be ready to use. This is normally prepared by the user using the editor, (and the TLOSS22CC USER'S MANUAL, if necessary) at a terminal.

c. Submit the command file, TLOSS22CC.COM, to the batch stream (or batch queue) using a command such as one of these:

```
SUBMIT/NAME=LTT25MAR/QUEUE=SLOW/AFTER=20:00 TLOSS22CC
        SLOW queue, Job starts at 8:00 P.M.

SUBMIT/NAME=LTT25MAR/QUEUE=SLOW TLOSS22CC
        SLOW queue, Job starts now

SUBMIT/NAME=LTT25MAR TLOSS22CC
        SYS$BATCH queue ( the 'regular' queue ),
        Job starts now
```

d. The following command file, also prepared using the editor, is a typical file submitted, using one of the previous submit commands.

```
$ASSIGN LTT25MAR SYS$PRINT                   (1)
$ASSIGN LTT25MAR SYS$OUTPUT                  (2)
$ASSIGN DRO:LTT25MAR.PLT FOR007              (3)
$ASSIGN DRO:LTT25MAR.OUT FOR006              (4)
$ASSIGN DRO:SHALLOW.DAE FOR005               (5)
$TYPE TLOSS22CC                              (6)
$COPY DRO:SHALLOW.DAE DRO:LTT25MAR.DAT       (7)
$RUN TLOSS22CC                               (8)
```

This run is called the LTT25MAR run. 'LTT' stands for LOSS TWENTY TWO C. '25MAR' indicates that this is the twenty-fifth run in March.

The first two lines in this command file put the log file on the disk instead of the printer.
The third line puts unit 7 output into file LTT25MAR.PLT.
The fourth line puts unit 6 output into file LTT25MAR.OUT.
The fifth line draws unit 5 input from SHALLOW.DAE.
The sixth line copies TLOSS22CC.LIS to LTT25MAR.LIS.
The seventh line copies SHALLOW.DAE to LTT25MAR.DAT.
The eighth line runs program TLOSS22CC.

A command file such as this yields a file group; each file having the same name but a different extension. These files are then treated collectively. They are printed together and stored on tape together. Such a group is, for example:

LTT25MAR.DAT	a copy of the data file
LTT25MAR.LIS	the program listing file
LTT25MAR.LOG	the log file
LTT25MAR.OUT	the output file
LTT25MAR.PLT	the numerical output file

The user normally gets a complete printout of the run by typing the command:

```
PRINT LTT25MAR.*;*
```

on the terminal.

3. Copying Runs to Tape (Archiving)

Periodically, it is necessary to clear the old runs off the disk and put them on tape. This is a very simple operation and proceeds as follows:

a. Initialize a tape. Give it a name that is somehow indicative of the present month. For example:

```
INITIALIZE/DENSITY=1600 MT1: MARAPR
```

This tape would have runs made in March and April. This step is not necessary if the user is adding runs to a tape that is already partly filled. In such a case, be careful not to initialize!

b. Mount the tape with the MOUNT command.

```
MOUNT MT1: MARAPR
```

This is not the same as physically mounting the tape on the tape drive, which is assumed to have taken place previously.

c. Copy all the run files from disk to tape with the COPY command. For example in this case:

```
COPY *MAR.*;* MT1:
COPY *APR.*;* MT1:
```

would copy all the March and all the April runs onto the tape mounted on tape drive 1.

d. For future reference, it is good to get a list of names of the files on the tape. This is accomplished as follows:

```
DIRECTORY/PRINT MT1:
```

The printout resulting from this command does not have the tape name on it, so the user should write this on immediately.
If, for some reason, this printout is lost, the tape can be remounted and the directory can be taken again. But this is no reason to be careless with the original directory printout.

e. Dismount the tape with the dismount command:

```
DISMOUNT MT1:
```

f. Delete the files on the disk that have just been copied to the tape. In this case:

```
DELETE *MAR.*;*
DELETE *APR.*;*
```

4. Distributing the Program by Tape

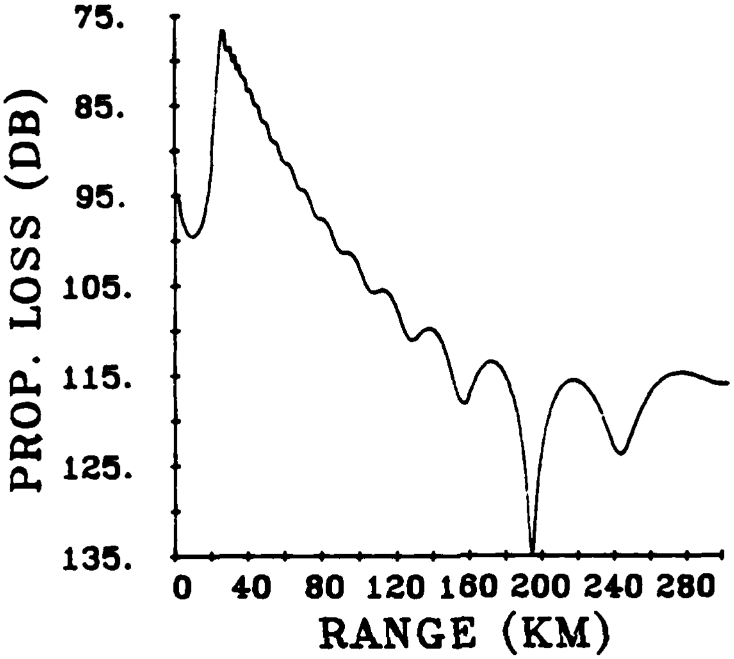
The program and test cases are distributed using tapes with the same format as the archival tapes. This is the Files-11 format, peculiar to Digital Equipment Corporation. The tapes from this system have a density of 1600 bpi, and a blocksize of 2048 bytes.

If any further questions arise, feel free to call:

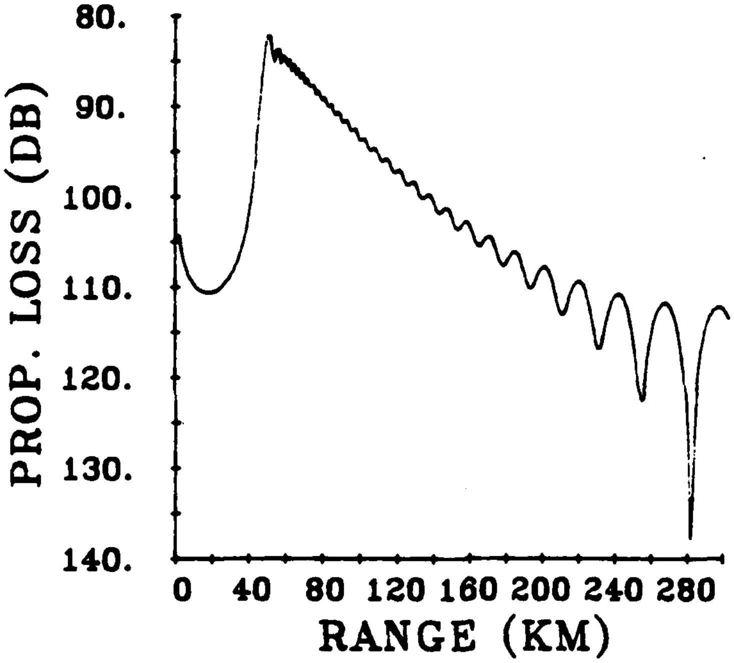
Dr. H. Kutschale	(914) 359-2900 x382
Mr. L. Carroll	(914) 359-2900 x221

```
                    TEST CASE DATA
05  1  0                       DATA FOR  TLOSS22CC
00.0      1500.0    0.000      0.00001   0.00000   1.00
  50.     1500.0    0.000      0.00001   0.00000   1.00
  75.     1500.0    0.000      0.00001   0.00000   1.00
4200.     1500.0    0.000      0.00001   0.00000   1.00
4200.     1600.0    0.000      0.00001   0.00000   1.00
   2    3    2 1024  -1     0  001  020.0     10000.0   1499.00   1600.01
0.0       00.0      0.00       00.00     0.000        0    1    1
05  1  0                       DATA FOR  TLOSS22CC
00.0      1500.0    0.000      0.00001   0.00000   1.00
  50.     1500.0    0.000      0.00001   0.00000   1.00
  75.     1500.0    0.000      0.00001   0.00000   1.00
4200.     1500.0    0.000      0.00001   0.00000   1.00
4200.     1600.0    0.000      0.00001   0.00000   1.00
   2    3    2 1024  -1     0  001  020.0     10000.0   1499.00   1600.01
0.0       00.0      0.00       00.00     0.000        0    0    2
05  1  0                       DATA FOR  TLOSS22CC
00.0      1500.0    0.000      0.00001   0.00000   1.00
  50.     1500.0    0.000      0.00001   0.00000   1.00
  75.     1500.0    0.000      0.00001   0.00000   1.00
4200.     1500.0    0.000      0.00001   0.00000   1.00
4200.     1600.0    0.000      0.00001   0.00000   1.00
   2    3    2 1024  -1     0  001  020.0     10000.0   1499.00   1600.01
0.0       00.0      0.00       00.00     0.000        0    1    3
05  1  0                       DATA FOR  TLOSS22CC
00.0      1500.0    0.000      0.00001   0.00000   1.00
  50.     1500.0    0.000      0.00001   0.00000   1.00
  75.     1500.0    0.000      0.00001   0.00000   1.00
4200.     1500.0    0.000      0.00001   0.00000   1.00
4200.     1600.0    0.000      0.00001   0.00000   1.00
   2    3    2 1024  00     0  001  020.0     10000.0   1499.00   1600.01
0.0       00.0      0.00       00.00     0.000        0    0    4
```

TRANSMISSION LS.

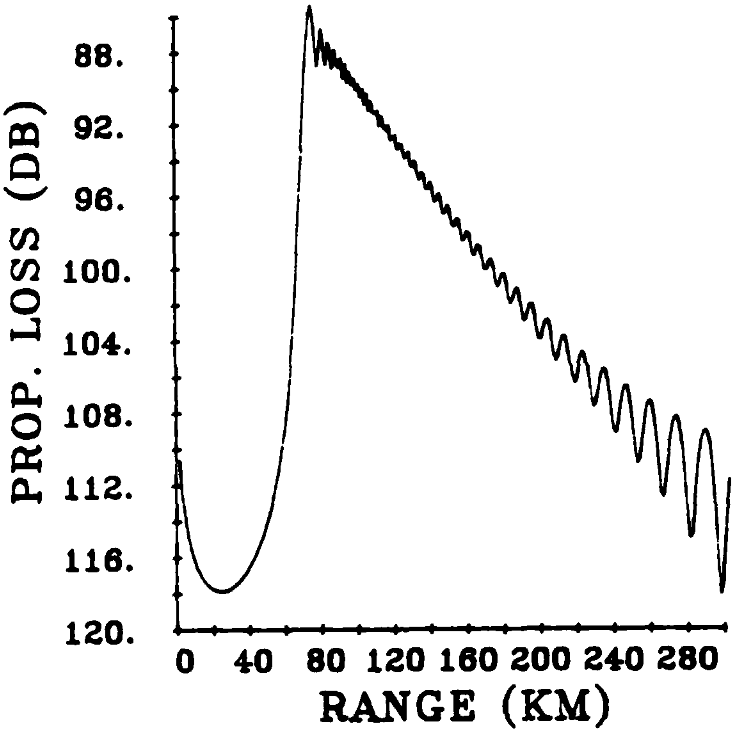


1 20.0 HZ L22C SM 0

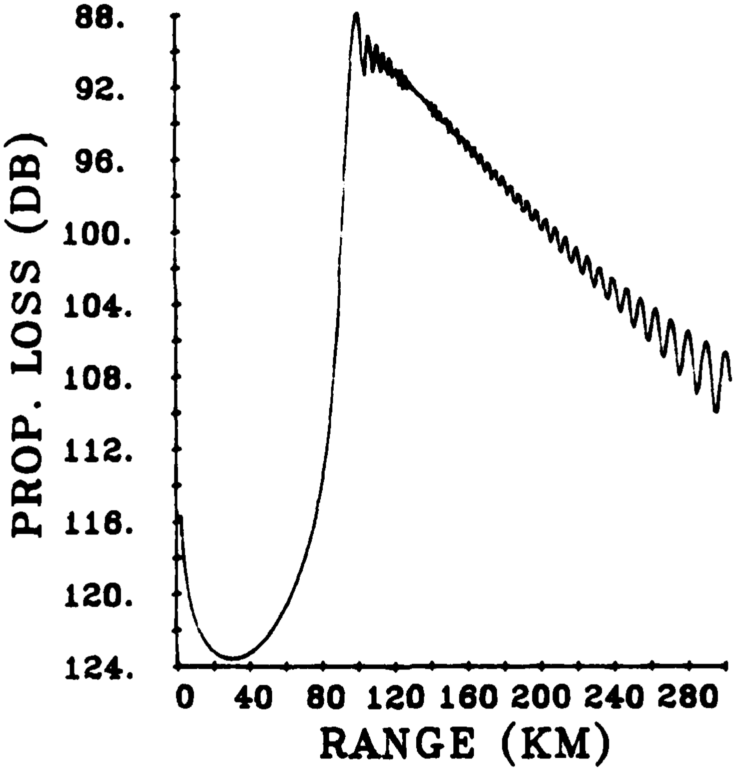


2 20.0 HZ L22C SM 0

TRANSMISSION LS.

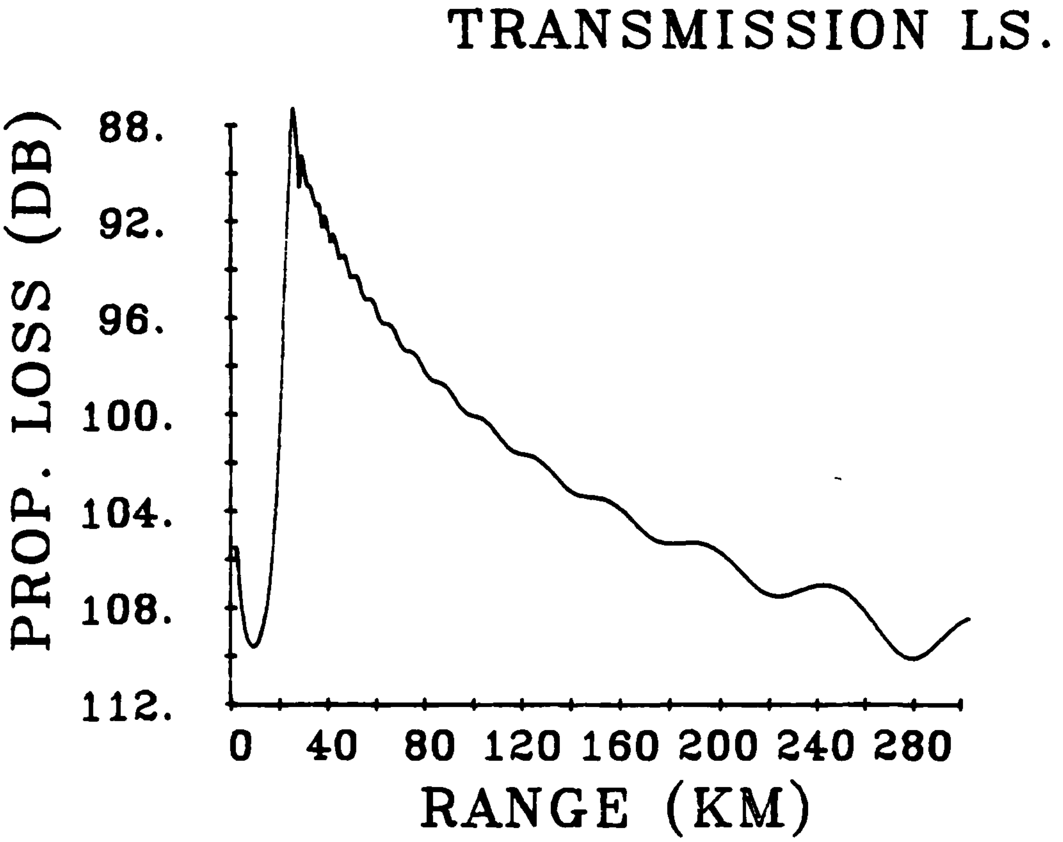


3 20.0 HZ L22C SM 0



4 20.0 HZ L22C SM 0

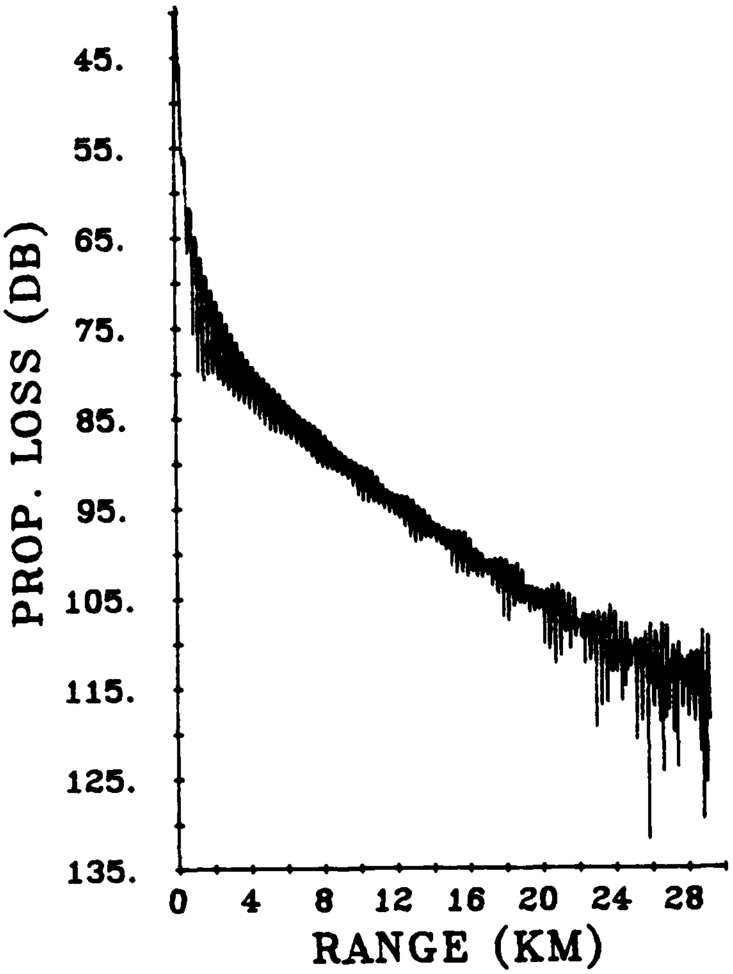
```
LTT38MAR.DAT;2                   6-APR-1983 15:31
                TEST CASE DATA
 04  1  0                       DATA FOR TLOSS22CC
 00.0      1500.0    0.0C0     0.00001   0.00000    1.00
 00.0      1500.0    0.000     0.00001   0.00000    1.00
 4125.     1500.0    0.000     0.00001   0.00000    1.00
 4125.     1600.0    0.000     0.00001   0.00000    1.00
   2     2    2 1024  00      0  001 020.0      10000.0   1499.00   1600.01
 0.0       1500.0    0.00001   01.00     25.00          0    1    0
```



1 20.0 HZ L22C SM 0

```
LTT39MAR.DAT;3                      6-APR-1983 15:31
                    TEST CASE DATA
                                                   DATA FOR TLOSS22CC
 06  1
  0.00      1500.0                .0000000                   1.00
 50.00      1500.0                .0000000                   1.00
 100.0      1500.0                .0000000                   1.00
 100.0      3000.0    1400.0      .00006286   .0001281 2.50
 125.0      3000.0    1400.0      .00006286   .0001281 2.50
 125.0      3000.0    1400.0      .00006286   .0001281 2.50
     2    3    02 1024    00   000    1    020.0    10000.       1000.0     8000.0
   0.0       1500.0   .0000000    1.00     0000.0         0      0   00      0
```

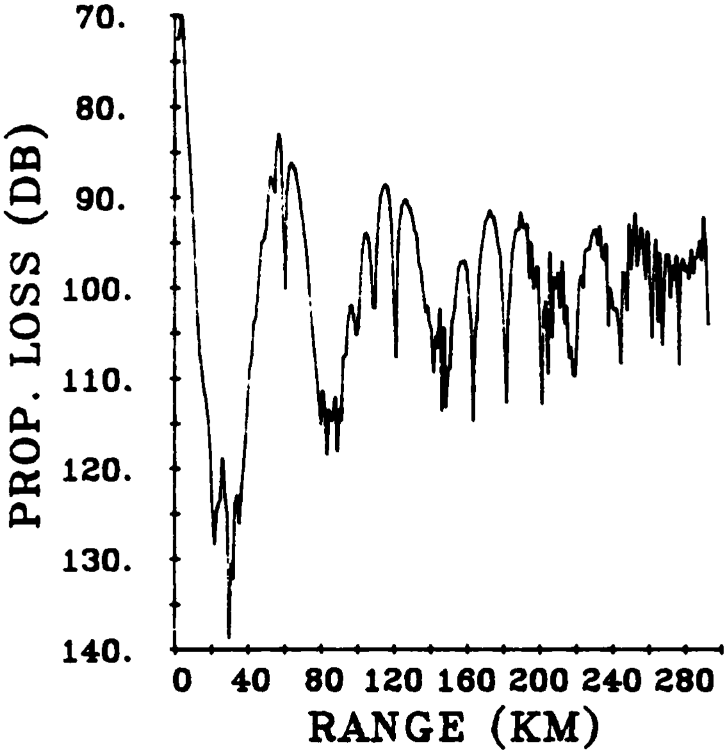
TRANSMISSION LS.



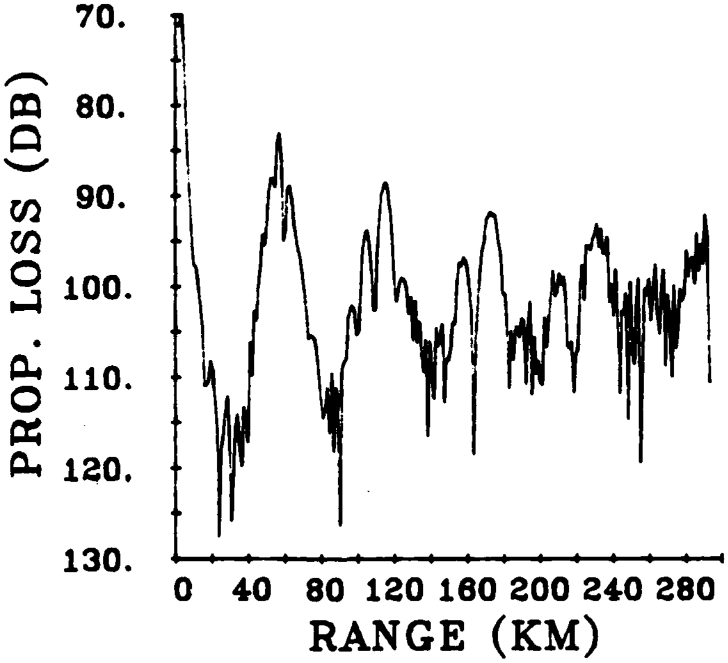
1 20.0 HZ L22C SM 0

```
LTT40MAR.DAT;3                        6-APR-1983 15:31
                   TEST CASE DATA

08   1
000.0       1536.5                 .0000100                1.0
152.4       1539.2                 .0000100                1.0
253.9       1530.0                 .0000100                1.0
406.3       1501.1                 .0000100                1.0
863.5       1475.0                 .0000100                1.0
1015.9      1471.9                 .0000100                1.0
5587.9      1549.6                 .0000100                1.0
5587.9      1549.6                 .0000100                1.9
    3    5     2   512    01    00      1   25.      10000.0      1470.0     1549.4
 00.0        0000.0      0.00000    0.0         000.0         0     0
    3    5     2   512    00    00      1   25.      10000.0      1470.0     1549.4
 00.0        1536.5      0.00001    1.0         000.0         0     0
```

TRANSMISSION LS.



1 25.0 HZ L22C SM 0



2 25.0 HZ L22C SM 0

TLOSS10HH USER'S MANUAL

A. Introduction

TLOSS10HH, our transmission loss (hereafter abbreviated as TL) program, was completed in June of 1982.

TLOSS10HH has these features and options for the user:

1. A choice of computational methods;
   a. The FFP method
   b. The Normal Mode method

2. Optional horizontal or vertical displacement in addition to the usual pressure values for TL.

3. A comprehensive selection of modelling parameters. These include:
   a. A detailed sound velocity profile
   b. Choice of source depth
   c. Choice of detector depth
   d. Choice of frequency
   e. Choice of wave number integrand limits
   f. Optional beaming, as it is affected by integrand tapering (in the FFP mode)
   g. Optional surface roughness
   h. Choice of either fluid or vacuum upper half space
   i. Possibility of the source and detector in the fluid upper half space
   j. Optional internal roughness
   k. Choice of initial range, range step, and range count (for the Normal Mode method)

4. Output. This includes:
   a. A printout of all input
   b. The TL plot
   c. The wave integrand plot of the FFP; either short or long.
   d. A table of the denominator values for the Normal Mode method
   e. A table of roots and their corresponding phase velocities, slopes, and excitation functions for the Normal Mode method

   Note: All short plots except the wave integrands are repeated, the repeated plot being a smoothed copy of the first and having the same bounds as the first.

5. Additional Optional Tables and Plots. These include:
   a. The TL plot in nautical miles, if desired and if the range is sufficiently long. The original TL plot is in kilometers.
   b. The denominator plot for the Normal Mode method.

6. Numerical output of the TL for further processing or plotting. Formerly, this was called punched output, since it went out to the card punch.

7. A profile check for multiple ducts. This check is made for the Normal Mode method. If multiple ducts are present, special care is taken to locate those 'hidden' roots which are only found by reducing the step size of the denominator function.

B. Input Variables and Formats

Program input for TLOSS10HH includes both the physical model and program control. This section of the manual will list and explain each line of the input.

1. Entry 1, one line

```
MMAXF, NCARD                                  variable names
( 2I3 )                                       format
```

MMAXF is a control variable which is used to read in the correct number of points in the sound velocity profile. (see 2. - next Entry)

NCARD is a control variable that, when set non-zero allows the program to put out TL data on unit 7 in a form which is readable as input by another program. A zero value will prevent any output on unit 7. This is normally used to make graphic plots. (Originally, on the IBM 360/91, unit 7 was the punched card unit, hence the name NCARD.)

2. Entry 2, MMAXF lines (max. of 49 lines)

```
(( V(M,N), N=1,7 ), M=1,MMAXF )               variable name
( 7D10.0 )                                    format
```

V is an array that contains the sound velocity profile. For each point in the profile, and on each line, (one point per line), the program expects:

a. a depth value, in meters
b. a sound (pressure) velocity value, in meters per second
c. a sound (shear) velocity value, in meters per second, if applicable
d. a corresponding attenuation value for pressure
e. a corresponding attenuation value for shear, if applicable
f. a density value, in grams per c.c., or specific density
g. a roughness value, if applicable

Note: No depth value can be less than a previous depth value. The program will stop if this is so.

3. Entry 3, one line

```
KSF, LRF, MM, LS, KKK, IBEAM, IMAX, FREQ, PO, C, CMAX
                                                       variable name
( 7I5, 4D10.0 )                                        format
```

KSF is a pointer to the source depth value. It points to one of the points in the sound velocity profile; the one which is the source depth value.

LRF is a pointer to the detector (or receiver) depth value. It's function is similar to KSF.

MM is a TL plot pruning factor, with a minimum acceptable value of 2. MM is applicable to FFP only. If an MM value of N is read in, then the first 1/N of the plot will be plotted. This is useful either to get a more detailed look at the output or to prune away that part of the plot affected by range aliasing.

LS is the number of points used in the wave integration. It also affects the range of the TL plot, for the FFP method.

KKK is a control variable that indicates:

a. If zero, this is the last case.
b. If negative, the next case starts with Entry 1, (new profile).
c. If positive, the next case starts with Entry 3.

IBEAM is a control variable that, when set non-zero, requires the program to taper the wave integrand at each end.

IMAX is the range count or the number of points in the TL plot for a Normal Mode case.

FREQ is the frequency for this case, in Hertz.

PO is the source level. For TL PO = 1.0

C is the lower bound sound velocity in the wave integration, in meters per second. For the Normal Mode method, if a zero is entered, a default value will be found from the sound velocity profile.

CMAX is the upper bound sound velocity in the wave integration, in meters per second. If a zero is entered here, the wave number integration will start from zero for the FFP method, or, for the Normal Mode method, a default value will be found from the sound velocity profile.

4. Entry 4, one line

```
REFLTO, ALPHAO, ATTENO, RHOO, H, INAUT, LONG, KHPV, IDPL
                                                           variable names
( 5D10.0, 4I5 )                                            format
```

REFLTO is the RMS surface roughness value, in meters. If this value is non-zero, the program explicitly sets the next three values to zero.

ALPHAO is, if present, the sound velocity in the fluid upper half space, in meters per second. This is the variable that determines whether the present case has either a fluid or vacuum upper half space.

ATTENO is, if present, the attenuation in the fluid upper half space.

RHOO is, if present, the density of the fluid upper half space, in grams per c.c.

H is, if present, the distance between the source and detector in the fluid upper half space. To use this option:

a. The first two sound velocity depths must be zero.
b. There must be some variation in the pressure sound velocity profile.
c. There must not be any shear sound velocity.
d. ALPHAO and H must be present.

INAUT is a control variable that, when set non-zero, allows the plotting of the TL plot in nautical miles if the range is greater than 100 kilometers. This is for the FFP method only.

LONG is a control variable that, when set non-zero, switches the wave integrand plot from short to long. This is used when it is necessary to examine each point of the wave integrand. FFP only.

KHPV is the control variable that determines whether output will be:

a. pressure, if zero.
b. vertical displacement, if positive.
c. horizontal displacement, if negative.

IDPL is the control variable that, when set, allows the denominator function to be plotted. Normal Mode method only.

5. Entry 5, one line

```
RA, DLRA                                                variable names
( 2D10.0 )                                              format
```

RA is the initial range of the TL plot, in meters.

DLRA is the range step of the TL plot, in meters.

These two variables are read in for the Normal Mode method. Since the TL plot is in kilometers, it is very important to be careful with the units of these two variables.

Note:

The program determines the computational method, (whether FFP or Normal Mode), by examining all of the attenuation values and the surface roughness value. If they are all zero, the Normal Mode method is selected.

C. Operating Procedures

The primary computer at Lamont-Doherty Geological Observatory is a VAX 11/780, which runs under the VMS operating system. Listed below are four distinctly different operations that are used to process TLOSS10HH.

1. Compiling-Linking
   a. Compiling (FORT command)
   Compiling the TLOSS10HH program on the VAX 11/780 has always been done with the following switches on:
      1. /CHECK Forbids array pointer overflow.
      2. /NOI4 Integer variables are set to 2 bytes.
      3. /NOF77 Where the compiler needs to choose between Fortran 77 and Fortran 4+, the latter is chosen.
      4. /LIST A compiler listing is made and stored on the disk as TLOSS10HH.LIS.

   b. Linking (LINK command)
   Linking is straight-forward. No extra routines are linked with TLOSS10HH.

2. Running
   TLOSS10HH is normally run through a batch stream, in particular the SLOW batch stream, since this is the cheapest route, especially at night. However, there is another batch stream with higher priority (and higher cost) that can be used for quicker results, the SYS$BATCH batch stream. In order to use either of these two batch streams:

   a. The executable file (TLOSS10HH.EXE) must be ready to use. The LINK operation satisfies this requirement.

   b. The data file must also be ready to use. This is normally prepared by the user using the editor, (and the TLOSS10HH USER'S MANUAL, if necessary) at a terminal.

   c. Submit the command file, TLOSS10HH.COM, to the batch stream (or batch queue) using a command such as one of these:

```
SUBMIT/NAME=LTH22JUN/QUEUE=SLOW/AFTER=20:00 TLOSS10HH
        SLOW queue, Job starts at 8:00 P.M.

SUBMIT/NAME=LTH22JUN/QUEUE=SLOW TLOSS10HH
        SLOW queue, Job starts now

SUBMIT/NAME=LTH22JUN TLOSS10HH
        SYS$BATCH queue ( the 'regular' queue ),
        Job starts now
```

   d. The following command file, also prepared using the editor, is a typical file submitted, using one of the previous submit commands.

```
$ASSIGN LTH22JUN SYS$PRINT                    (1)
$ASSIGN LTH22JUN SYS$OUTPUT                   (2)
$ASSIGN DR0:LTH22JUN.PLT FOR007               (3)
$ASSIGN DR0:LTH22JUN.OUT FOR006               (4)
$ASSIGN DR0:STANDARD.DAF FOR005               (5)
$TYPE TLOSS10HH                               (6)
$COPY DR0:STANDARD.DAF DR0:LTH22JUN.DAT       (7)
$RUN TLOSS10HH                                (8)
```

This run is called the LTH22JUN run. 'LTH' stands for LOSS TEN H. '22JUN' indicates that this is the 22nd run in June.

The first two lines in this command file put the log file on the disk instead of the printer.
The third line puts unit 7 output into file LTH22JUN.PLT.
The fourth line puts unit 6 output into file LTH22JUN.OUT.
The fifth line draws unit 5 input from STANDARD.DAF.
The sixth line copies TLOSS10HH.LIS to LTH22JUN.LIS.
The seventh line copies STANDARD.DAF to LTH22JUN.DAT.
The eighth line runs program TLOSS10HH.

A command file such as this yields a file group; each file having the same name but a different extension. These files are then treated collectively. They are printed together and stored on tape together. Such a group is, for example:

LTH22JUN.DAT	a copy of the data file
LTH22JUN.LIS	the program listing file
LTH22JUN.LOG	the log file
LTH22JUN.OUT	the output file
LTH22JUN.PLT	the numerical output file

The user normally gets a complete printout of the run by typing the command:

```
PRINT LTH22JUN.*;*
```

on the terminal.

3. Copying Runs to Tape (Archiving)

Periodically, it is necessary to clear the old runs off the disk and put them on tape. This is a very simple operation and proceeds as follows:

a. Initialize a tape. Give it a name that is somehow indicative of the present month. For example:

```
INITIALIZE/DENSITY=1600 MT1: JUNJUL
```

This tape would have runs made in June and July. This step is not necessary if the user is adding runs to a tape that is already partly filled. In such a case, be careful not to initialize!

b. Mount the tape with the MOUNT command.

```
MOUNT MT1: JUNJUL
```

This is not the same as physically mounting the tape on the tape drive, which is assumed to have taken place previously.

c. Copy all the run files from disk to tape with the COPY command. For example in this case:

```
COPY *JUN.*;* MT1:
COPY *JUL.*;* MT1:
```

would copy all the June and all the July runs onto the tape mounted on tape drive 1.

d. For future reference, it is good to get a list of names of the files on the tape. This is accomplished as follows:

```
DIRECTORY/PRINT MT1:
```

The printout resulting from this command does not have the tape name on it, so the user should write this on immediately.
If, for some reason, this printout is lost, the tape can be remounted and the directory can be taken again. But this is no reason to be careless with the original directory printout.

e. Dismount the tape with the dismount command:

```
DISMOUNT MT1:
```

f. Delete the files on the disk that have just been copied to the tape. In this case:

```
DELETE *JUN.*;*
DELETE *JUL.*;*
```

4. Distributing the Program by Tape

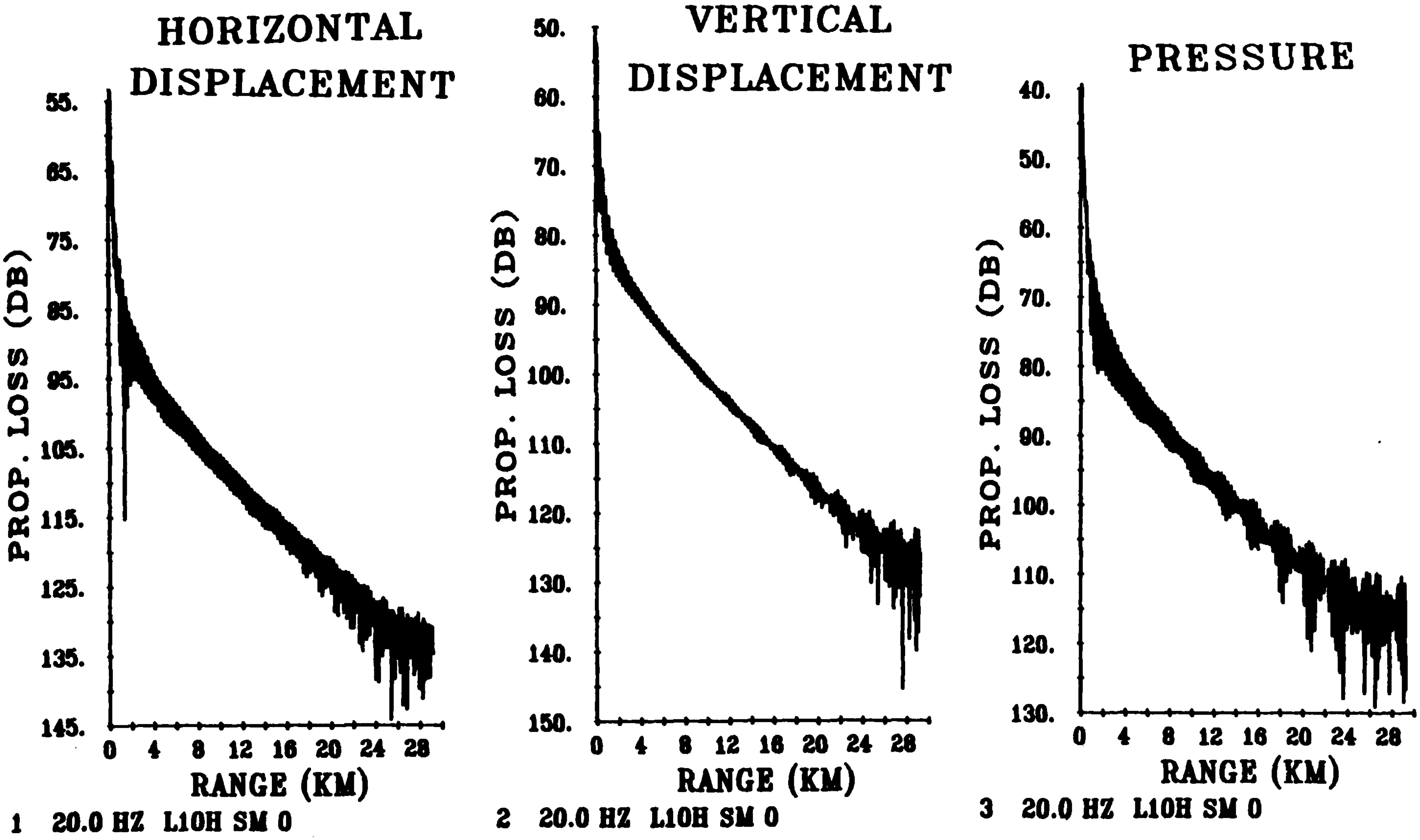
The program and test cases are distributed using tapes with the same format as the archival tapes. This is the Files-11 format, peculiar to Digital Equipment Corporation. The tapes from this system have a density of 1600 bpi, and a blocksize of 2048 bytes.

If any further questions arise, feel free to call:

Dr. H. Kutschale	(914) 359-2900 x382
Mr. L. Carroll	(914) 359-2900 x221

```
LTH01OCT.DAT;2                     6-APR-1983 15:31
                     TEST CASE DATA
 06  1                            DATA FOR TLOSS10HH
 0.00    1500.0          .0000150           1.00
50.00    1500.0          .0000150           1.00
100.0    1500.0          .0000150           1.00
100.0    3000.0  1400.0  .00007700 .0001600 2.50
500.0    3000.0  1400.0  .00007700 .0001600 2.50
500.0    3000.0  1400.0  .00007700 .0001600 2.50
   2   3  02 1024  -1 000   1  020.0  10000.  1000.0  8000.0
 0.0      1500.0 .0000150  1.00   0000.0    0   1  -1    1
 00.      00.0
 06  1                            DATA FOR TLOSS10HH
 0.00    1500.0          .0000150           1.00
50.00    1500.0          .0000150           1.00
100.0    1500.0          .0000150           1.00
100.0    3000.0  1400.0  .00007700 .0001600 2.50
500.0    3000.0  1400.0  .00007700 .0001600 2.50
500.0    3000.0  1400.0  .00007700 .0001600 2.50
   2   3  02 1024  -1 000   1  020.0  10000.  1000.0  8000.0
 0.0      1500.0 .0000150  1.00   0000.0    0   1  01    1
 00.      00.0
 06  1                            DATA FOR TLOSS10HH
 0.00    1500.0          .0000150           1.00
50.00    1500.0          .0000150           1.00
100.0    1500.0          .0000150           1.00
100.0    3000.0  1400.0  .00007700 .0001600 2.50
500.0    3000.0  1400.0  .00007700 .0001600 2.50
500.0    3000.0  1400.0  .00007700 .0001600 2.50
   2   3  02 1024  00 000   1  020.0  10000.  1000.0  8000.0
 0.0      1500.0 .0000150  1.00   0000.0    0   1  00    1
 00.      00.0
```

TRANSMISSION LS.



TLOSS17HH USER'S MANUAL

A. Introduction

TLOSS17HH, our more complete transmission loss (hereafter abbreviated as TL) program, was completed in June of 1982. It is an extension of TLOSS10HH, the interim standard TL program.

TLOSS17HH has these features and options for the user:

1. A choice of computational methods;
   a. The FFP method
   b. The Normal Mode method

   If the Normal Mode method is chosen, the user also has the option of including either of two branch cut integrals.

2. Optional horizontal or vertical displacement in addition to the usual pressure values for TL.

3. A comprehensive selection of modelling parameters. These include:
   a. A detailed sound velocity profile
   b. Choice of source depth
   c. Choice of detector depth
   d. Choice of frequency
   e. Choice of wave number integrand limits
   f. Optional beaming, as it is affected by integrand tapering (in the FFP mode)
   g. Optional surface roughness
   h. Choice of either fluid or vacuum upper half space
   i. Possibility of the source and detector in the fluid upper half space
   j. Optional internal roughness
   k. Choice of initial range, range step, and range count (for the Normal Mode method)

4. Output. This includes:
   a. A printout of all input
   b. The TL plot
   c. The wave integrand plot of the FFP; either short or long.
   d. A table of the denominator values for the Normal Mode method
   e. A table of roots and their corresponding phase velocities, slopes, and excitation functions for the Normal Mode method

   Note: All short plots except the wave integrands are repeated, the repeated plot being a smoothed copy of the first and having the same bounds as the first.

5. Additional Optional Tables and Plots. These include:
   a. The TL plot in nautical miles, if desired and if the range is sufficiently long. The original TL plot is in kilometers.
   b. The denominator plot for the Normal Mode method.
   c. Modal TL plots, Branch TL plots, and Total TL plots, if a branch cut option is chosen.

6. Numerical output of the TL for further processing or plotting. Formerly, this was called punched output, since it went out to the card punch.

7. A profile check for multiple ducts. This check is made for the Normal Mode method. If multiple ducts are present, special care is taken to locate those 'hidden' roots which are only found by reducing the step size of the denominator function.

B. Input Variables and Formats

Program input for TLOSS17HH includes both the physical model and program control. This section of the manual will list and explain each line of the input.

1. Entry 1, one line

```
MMAXF, NCARD                                    variable names
( 2I3 )                                         format
```

MMAXF is a control variable which is used to read in the correct number of points in the sound velocity profile. (see 2. - next Entry)

NCARD is a control variable that, when set non-zero allows the program to put out TL data on unit 7 in a form which is readable as input by another program. A zero value will prevent any output on unit 7. This is normally used to make graphic plots. (Originally, on the IBM 360/91, unit 7 was the punched card unit, hence the name NCARD.)

2. Entry 2, MMAXF lines (max. of 49 lines)

```
(( V(M,N), N=1,7 ), M=1,MMAXF )                 variable name
( 7D10.0 )                                      format
```

V is an array that contains the sound velocity profile. For each point in the profile, and on each line, (one point per line), the program expects:
a. a depth value, in meters
b. a sound (pressure) velocity value, in meters per second
c. a sound (shear) velocity value, in meters per second, if applicable
d. a corresponding attenuation value for pressure
e. a corresponding attenuation value for shear, if applicable
f. a density value, in grams per c.c., or specific density
g. a roughness value, if applicable

Note: No depth value can be less than a previous depth value. The program will stop if this is so.

3. Entry 3, one line

```
KSF, LRF, MM, LS, KKK, IBEAM, IMAX, FREQ, PO, C, CMAX
                                                variable names
( 7I5, 4D10.0 )                                 format
```

KSF is a pointer to the source depth value. It points to one of the points in the sound velocity profile; the one which is the source depth value.

LRF is a pointer to the detector (or receiver) depth value. It's function is similar to KSF.

MM is a TL plot pruning factor, with a minimum acceptable value of 2. MM is applicable to FFP only. If an MM value of N is read in, then the first 1/N of the plot will be plotted. This is useful either to get a more detailed look at the output or to prune away that part of the plot affected by range aliasing.

LS is the number of points used in the wave integration. It also affects the range of the TL plot, for the FFP method.

KKK is a control variable that indicates:
a. If zero, this is the last case.
b. If negative, the next case starts with Entry 1, (new profile).
c. If positive, the next case starts with Entry 3.

IBEAM is a control variable that, when set non-zero, requires the program to taper the wave integrand at each end.

IMAX is the range count or the number of points in the TL plot for a Normal Mode case.

FREQ is the frequency for this case, in Hertz.

PO is the source level. For TL PO = 1.0

C is the lower bound sound velocity in the wave integration, in meters per second. For the Normal Mode method, if a zero is entered, a default value will be found from the sound velocity profile.

CMAX is the upper bound sound velocity in the wave integration, in meters per second. If a zero is entered here, the wave number integration will start from zero for the FFP method, or, for the Normal Mode method, a default value will be found from the sound velocity profile.

4. Entry 4, one line

REFLTO, ALPHAO, ATTENO, RHOO, H, INAUT, LONG, KHPV, IBR,
IDPL variable names
(5D10.0, 5I5) format

REFLTO is the RMS surface roughness value, in meters. If this value is non-zero, the program explicitly sets the next three values to zero.

ALPHAO is, if present, the sound velocity in the fluid upper half space, in meters per second. This is the variable that determines whether the present case has either a fluid or vacuum upper half space.

ATTENO is, if present, the attenuation in the fluid upper half space.

RHOO is, if present, the density of the fluid upper half space, in grams per c.c.

H is, if present, the distance between the source and detector in the fluid upper half space. To use this option:

a. The first two sound velocity depths must be zero.
b. There must be some variation in the pressure sound velocity profile.
c. There must not be any shear sound velocity.
d. ALPHAO and H must be present.

INAUT is a control variable that, when set non-zero, allows the plotting of the TL plot in nautical miles if the range is greater than 100 kilometers. This is for the FFP method only.

LONG is a control variable that, when set non-zero, switches the wave integrand plot from short to long. This is used when it is necessary to examine each point of the wave integrand. FFP only.

KHPV is the control variable that determines whether output will be:

a. pressure, if zero.
b. vertical displacement, if positive.
c. horizontal displacement, if negative.

IBR is the control variable that determines if a branch cut integral will be included with the Normal Mode method.

a. If zero, no.
b. If positive, yes (exact)
c. If negative, yes (FFP)

IDPL is the control variable that, when set, allows the denominator function to be plotted. Normal Mode method only.

5. Entry 5, one line

```
RA, DLRA                                          variable names
( 2D10.0 )                                        format
```

RA is the initial range of the TL plot, in meters.

DLRA is the range step of the TL plot, in meters.

These two variables are read in for the Normal Mode method, when IBR is zero or positive, only. Since the TL plot is in kilometers, it is very important to be careful with the units of these two variables.

Note:

The program determines the computational method, (whether FFP or Normal Mode), by examining all of the attenuation values and the surface roughness value. If they are all zero, the Normal Mode method is selected.

C. Operating Procedures

The primary computer at Lamont-Doherty Geological Observatory is a VAX 11/780, which runs under the VMS operating system. Listed below are four distinctly different operations that are used to process TLOSS17HH.

1. Compiling-Linking

a. Compiling (FORT command)
Compiling the TLOSS17HH program on the VAX 11/780 has always been done with the following switches on:

1. /CHECK Forbids array pointer overflow.
2. /NOI4 Integer variables are set to 2 bytes.
3. /NOF77 Where the compiler needs to choose between Fortran 77 and Fortran 4+, the latter is chosen.
4. /LIST A compiler listing is made and stored on the disk as TLOSS17HH.LIS.

b. Linking (LINK command)
Linking is straight-forward. No extra routines are linked with TLOSS17HH.

2. Running

TLOSS17HH is normally run through a batch stream, in particular the SLOW batch stream, since this is the cheapest route, especially at night. However, there is another batch stream with higher priority (and higher cost) that can be used for quicker results, the SYS$BATCH batch stream. In order to use either of these two batch streams:

a. The executable file (TLOSS17HH.EXE) must be ready to use. The LINK operation satisfies this requirement.

b. The data file must also be ready to use. This is normally prepared by the user using the editor, (and the TLOSS17HH USER'S MANUAL, if necessary) at a terminal.

c. Submit the command file, TLOSS17HH.COM, to the batch stream (or batch queue) using a command such as one of these:

```
SUBMIT/NAME=LSH22JUN/QUEUE=SLOW/AFTER=20:00 TLOSS17HH
          SLOW queue, Job starts at 8:00 P.M.

SUBMIT/NAME=LSH22JUN/QUEUE=SLOW TLOSS17HH
          SLOW queue, Job starts now

SUBMIT/NAME=LSH22JUN TLOSS17HH
          SYS$BATCH queue ( the 'regular' queue ),
          Job starts now
```

d. The following command file, also prepared using the editor, is a typical file submitted, using one of the previous submit commands.

```
$ASSIGN LSH22JUN SYS$PRINT                  (1)
$ASSIGN LSH22JUN SYS$OUTPUT                 (2)
$ASSIGN DRO:LSH22JUN.PLT FOR007             (3)
$ASSIGN DRO:LSH22JUN.OUT FOR006             (4)
$ASSIGN DRO:SIMPLE19.DAF FOR005             (5)
$TYPE TLOSS17HH                             (6)
$COPY DRO:SIMPLE19.DAF DRO:LSH22JUN.DAT     (7)
$RUN TLOSS17HH                              (8)
```

This run is called the LSH22JUN run. 'LSH' stands for LOSS SEVENTEEN H. '22JUN' indicates that this is the 22nd run in June.

The first two lines in this command file put the log file on the disk instead of the printer.
The third line puts unit 7 output into file LSH22JUN.PLT.
The fourth line puts unit 6 output into file LSH22JUN.OUT.
The fifth line draws unit 5 input from SIMPLE19.DAF.
The sixth line copies TLOSS17HH.LIS to LSH22JUN.LIS.
The seventh line copies SIMPLE19.DAF to LSH22JUN.DAT.
The eighth line runs program TLOSS17HH.

A command file such as this yields a file group; each file having the same name but a different extension. These files are then treated collectively. They are printed together and stored on tape together. Such a group is, for example:

LSH22JUN.DAT	a copy of the data file
LSH22JUN.LIS	the program listing file
LSH22JUN.LOG	the log file
LSH22JUN.OUT	the output file
LSH22JUN.PLT	the numerical output file

The user normally gets a complete printout of the run by typing the command:

```
PRINT LSH22JUN.*;*
```

on the terminal.

3. Copying Runs to Tape (Archiving)

Periodically, it is necessary to clear the old runs off the disk and put them on tape. This is a very simple operation and proceeds as follows:

a. Initialize a tape. Give it a name that is somehow indicative of the present month. For example:

```
INITIALIZE/DENSITY=1600 MT1: JUNJUL
```

This tape would have runs made in June and July. This step is not necessary if the user is adding runs to a tape that is already partly filled. In such a case, be careful not to initialize!

b. Mount the tape with the MOUNT command.

```
MOUNT MT1: JUNJUL
```

This is not the same as physically mounting the tape on the tape drive, which is assumed to have taken place previously.

c. Copy all the run files from disk to tape with the COPY command. For example in this case:

```
COPY *JUN.*;* MT1:
COPY *JUL.*;* MT1:
```

would copy all the June and all the July runs onto the tape mounted on tape drive 1.

d. For future reference, it is good to get a list of names of the files on the tape. This is accomplished as follows:

```
DIRECTORY/PRINT MT1:
```

The printout resulting from this command does not have the tape name on it, so the user should write this on immediately.
If, for some reason, this printout is lost, the tape can be remounted and the directory can be taken again. But this is no reason to be careless with the original directory printout.

e. Dismount the tape with the dismount command:

```
DISMOUNT MT1:
```

f. Delete the files on the disk that have just been copied to the tape. In this case:

```
DELETE *JUN.*;*
DELETE *JUL.*;*
```

4. Distributing the Program by Tape

The program and test cases are distributed using tapes with the same format as the archival tapes. This is the Files-11 format, peculiar to Digital Equipment Corporation. The tapes from this system have a density of 1600 bpi, and a blocksize of 2048 bytes.

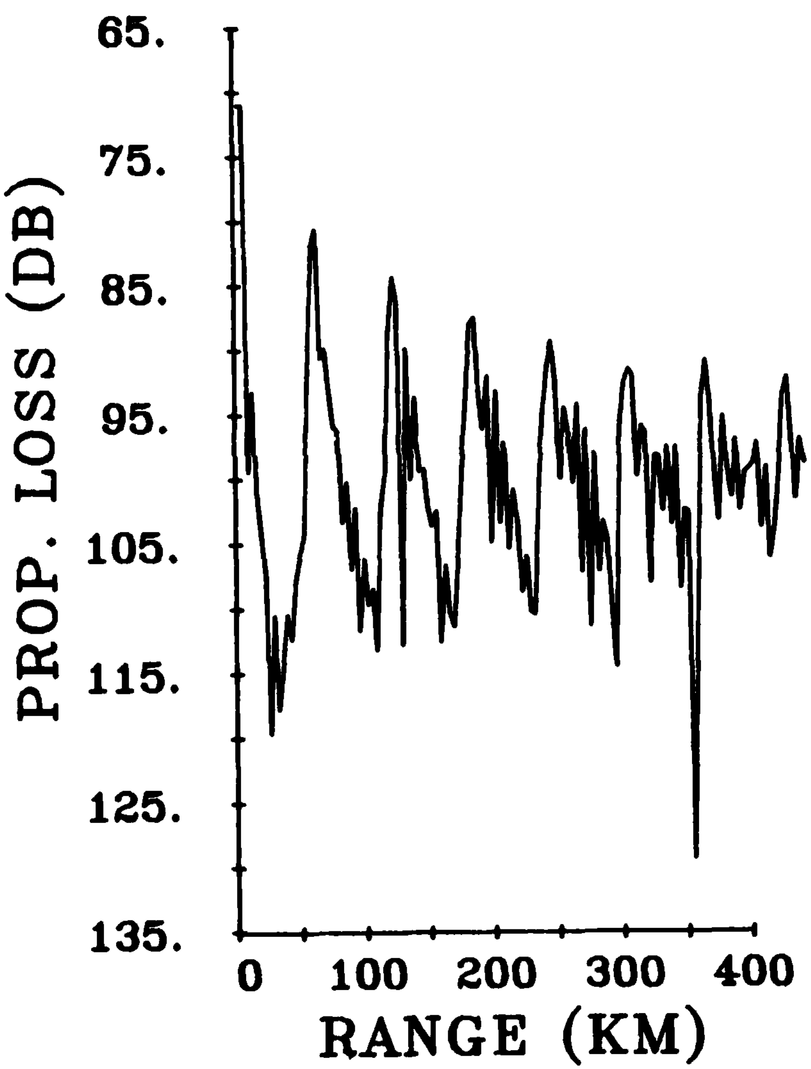
If any further questions arise, feel free to call:

Dr. H. Kutschale	(914) 359-2900 x382
Mr. L. Carroll	(914) 359-2900 x221

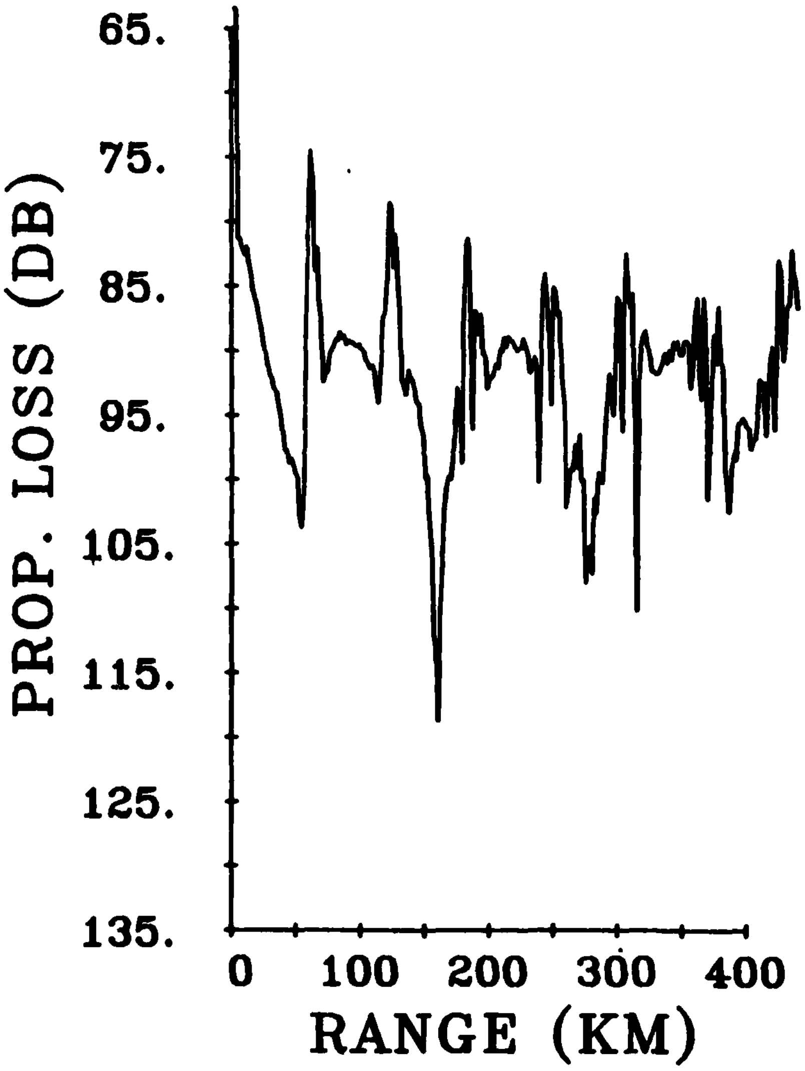
TEST CASE DATA

```
12  1  0                        DATA FOR TLOSS17HH
00.0      1494.0    0.000     0.00001   0.00000    1.03
61.0      1494.7    0.000     0.00001   0.00000    1.03
91.4      1495.2    0.000     0.00001   0.00000    1.03
330.0     1496.8    0.000     0.00001   0.00000    1.03
800.0     1487.9    0.000     0.00001   0.00000    1.03
1950.0    1495.6    0.000     0.00001   0.00000    1.03
3400.0    1511.7    0.000     0.00001   0.00000    1.03
3400.0    1511.7    0.000     0.00001   0.00000    1.03
3450.0    1511.7    0.000     0.00001   0.00000    1.03
3450.0    1511.7    0.000     0.00001   0.00000    1.03
3500.0    1511.7    0.000     0.00001   0.00000    1.03
3500.0    1511.7    0.000     0.00001   0.00000    1.03
   3    2    4 1024   01    0    1 025.0      10000.0   1485.0    1512.0
0.0       00.0      0.00      00.00     0.000         1    0
0.0       00.0
   3    2    2 1024   01    0    1 050.0      10000.0   1485.0    1512.0
0.0       00.0      0.00      00.00     0.000         1    0
0.0       00.0
   3    2    2 1024   01    0    1 075.0      10000.0   1485.0    1512.0
0.0       00.0      0.00      00.00     0.000         1    0
0.0       00.0
   3    2    2 1024   01    0    1 120.0      10000.0   1485.0    1512.0
0.0       00.0      0.00      00.00     0.000         1    0
0.0       00.0
   3    2    2 1024   00    0    1 160.0      10000.0   1485.0    1512.0
0.0       00.0      0.00      00.00     0.000         1    0
0.0       00.0
```

TRANSMISSION LS.

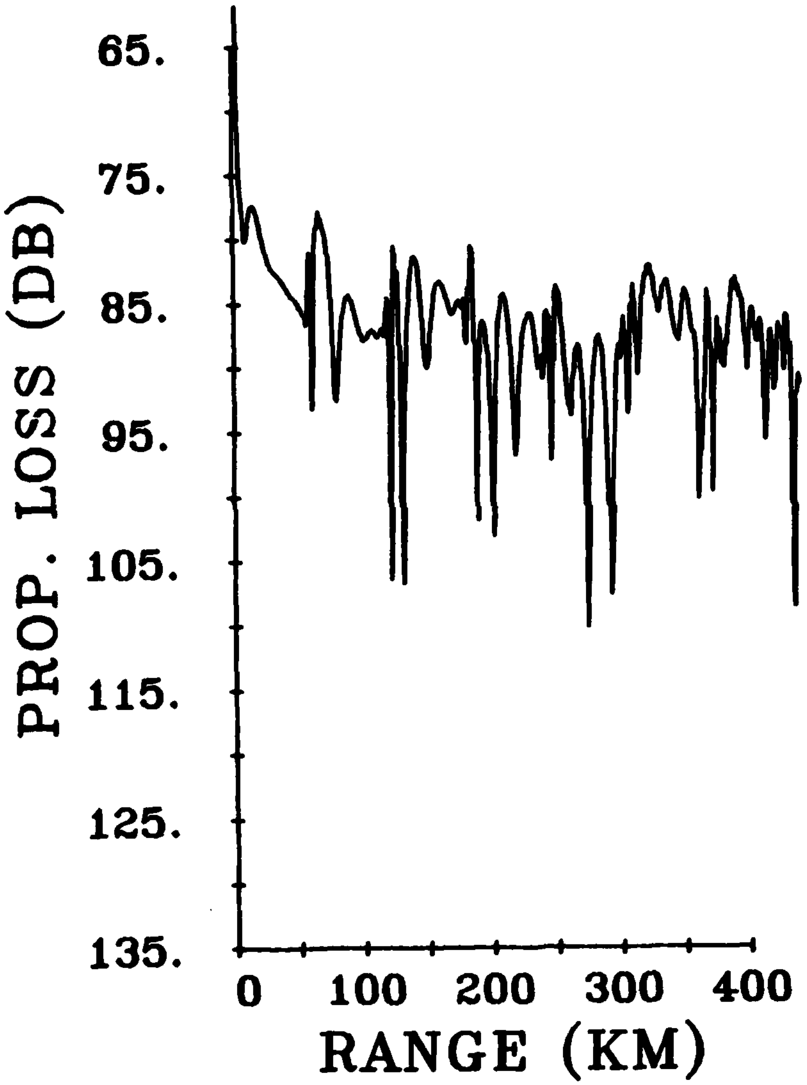


1 25.0 HZ L17H SM 0

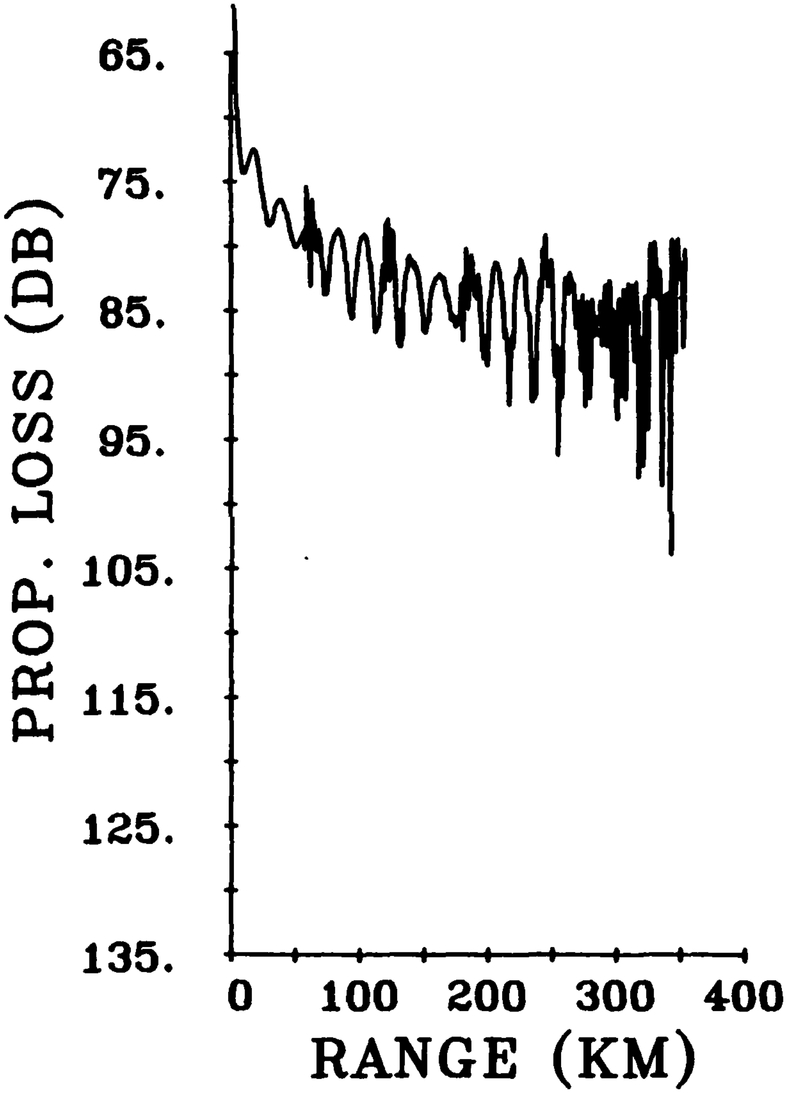


2 50.0 HZ L17H SM 0

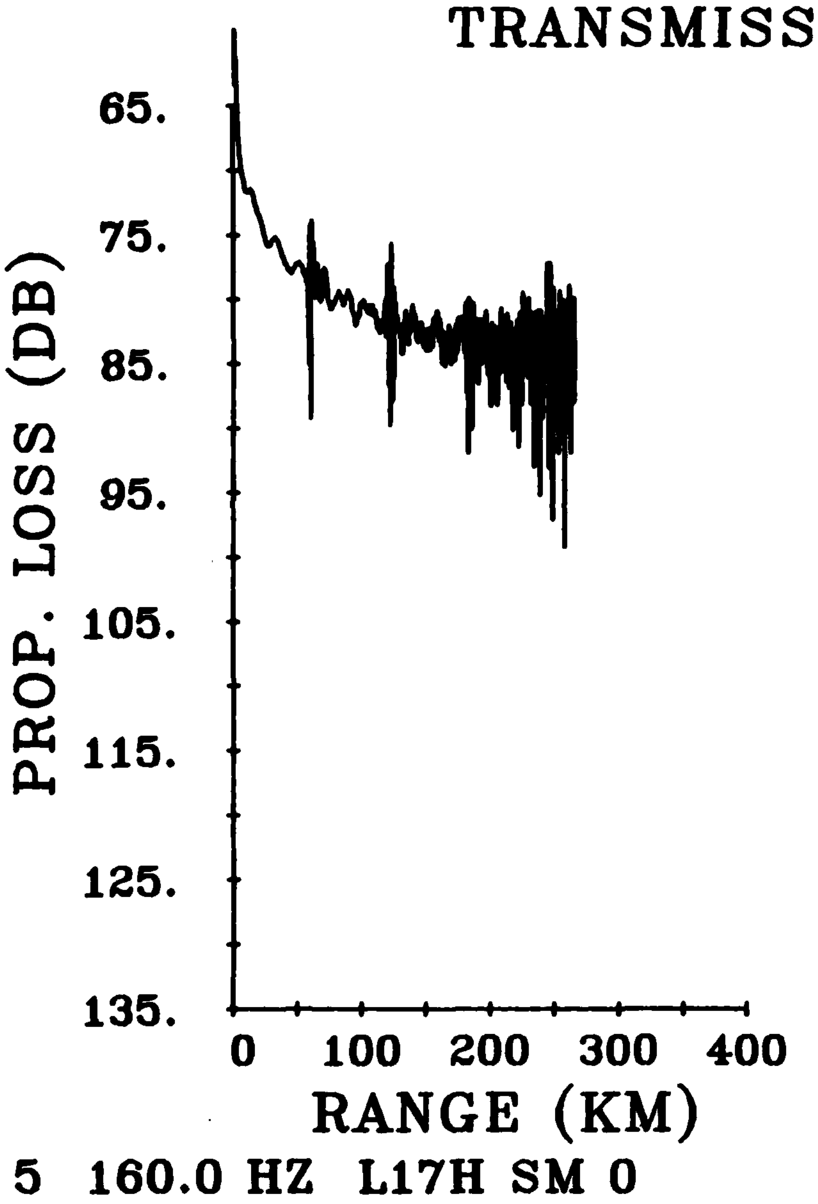
TRANSMISSION LS.



3 75.0 HZ L17H SM 0

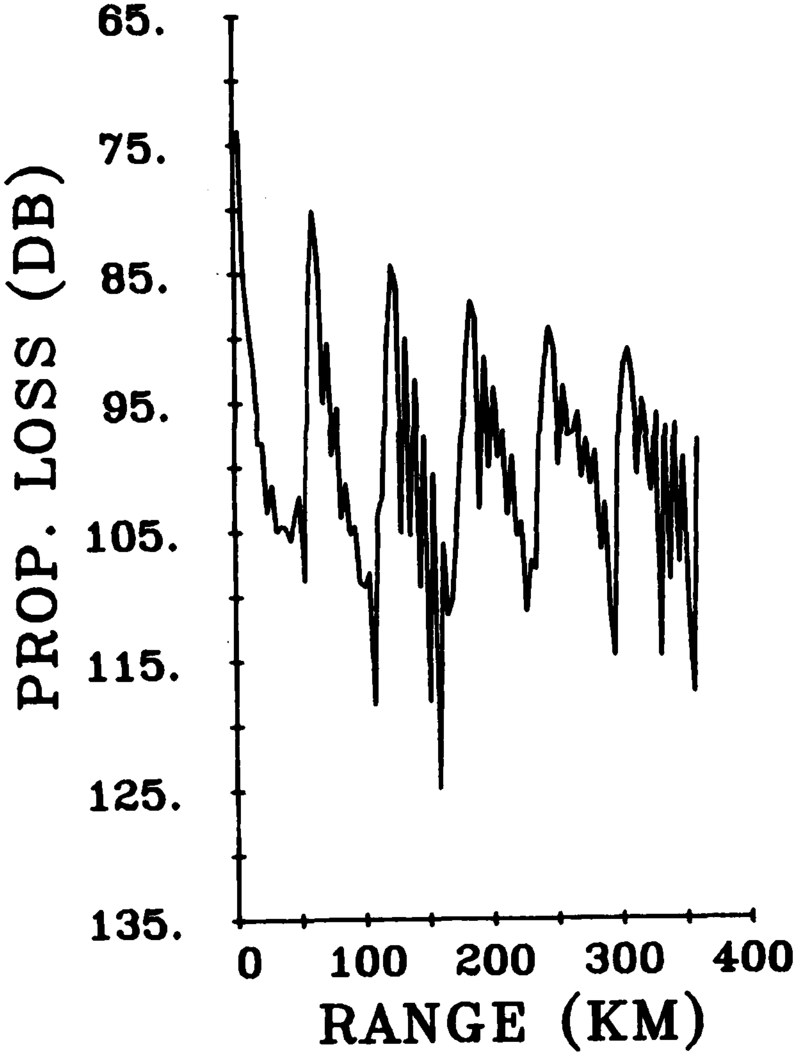


4 120.0 HZ L17H SM 0

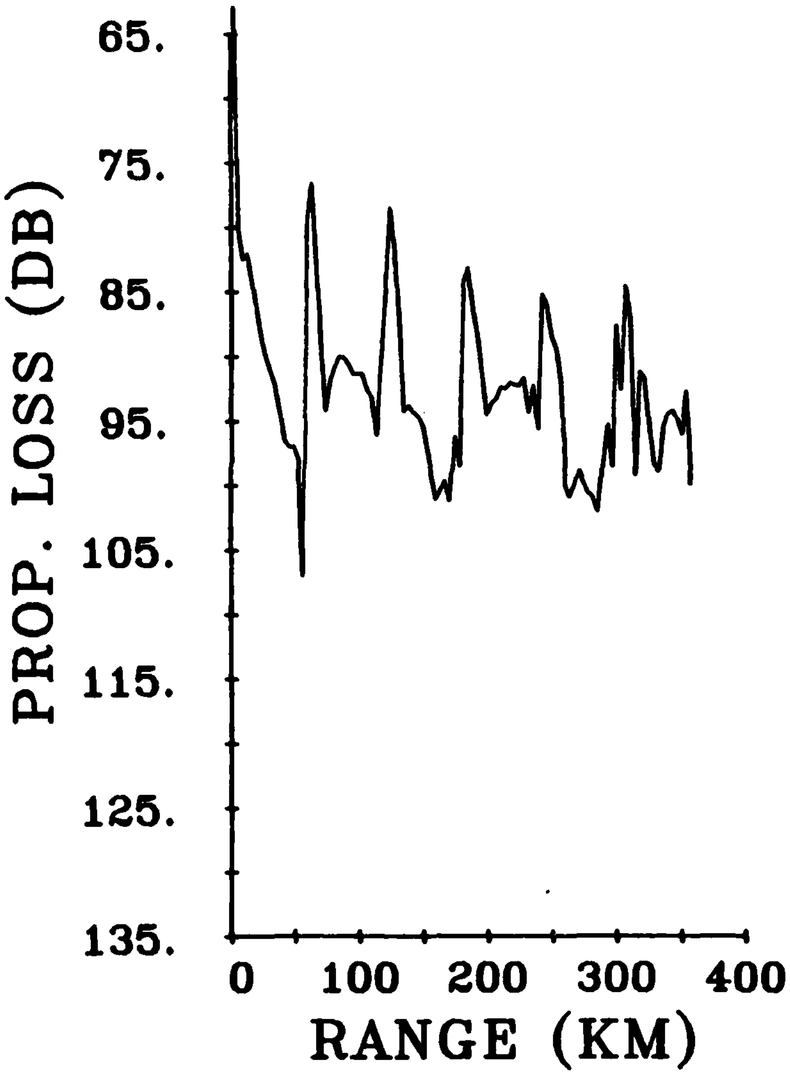

TRANSMISSION LS.
PROP. LOSS (DB)
65.
75.
85.
95.
105.
115.
125.
135.
0
100
200
300
400
RANGE (KM)
5 160.0 HZ L17H SM 0

```
LSH14JUL.DAT;2                        6-APR-1983 15:30
                         TEST CASE DATA
12  1  0                       DATA FOR TLOSS17HH
00.0      1494.0    0.000     0.00000   0.00000   1.03
61.0      1494.7    0.000     0.00000   0.00000   1.03
91.4      1495.2    0.000     0.00000   0.00000   1.03
330.0     1496.8    0.000     0.00000   0.00000   1.03
800.0     1487.9    0.000     0.00000   0.00000   1.03
1950.0    1495.6    0.000     0.00000   0.00000   1.03
3400.0    1511.7    0.000     0.00000   0.00000   1.03
3400.0    1511.7    0.000     0.00000   0.00000   1.03
3450.0    1511.7    0.000     0.00000   0.00000   1.03
3450.0    1511.7    0.000     0.00000   0.00000   1.03
3500.0    1511.7    0.000     0.00000   0.00000   1.03
3500.0    1511.7    0.000     0.00000   0.00000   1.03
   3    2    4 1024   01     0    1 025.0     10000.0   0000.0    00.000
0.0       00.0      0.00      00.00     0.000        1    0
0.0       3600.0
   3    2    2 1024   01     0    1 050.0     10000.0   0000.0    00.000
0.0       00.0      0.00      00.00     0.000        1    0
0.0       3600.0
   3    2    2 1024   01     0    1 075.0     10000.0   0000.0    00.000
0.0       00.0      0.00      00.00     0.000        1    0
0.0       3600.0
   3    2    2 1024   01     0    1 120.0     10000.0   0000.0    00.000
0.0       00.0      0.00      00.00     0.000        1    0
0.0       3600.0
   3    2    2 1024   00     0    1 160.0     10000.0   0000.0    00.000
0.0       00.0      0.00      00.00     0.000        1    0
0.0       3600.0
```

MODAL TRANS LOSS

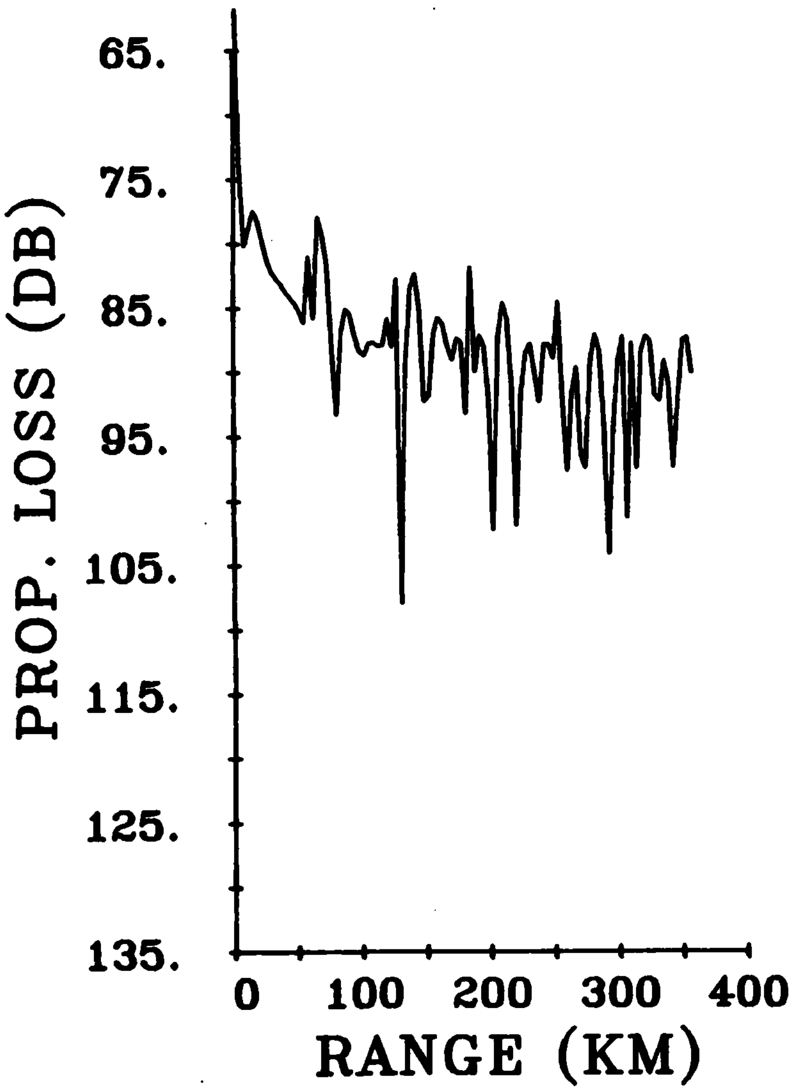


1 25.0 HZ L17H SM 0

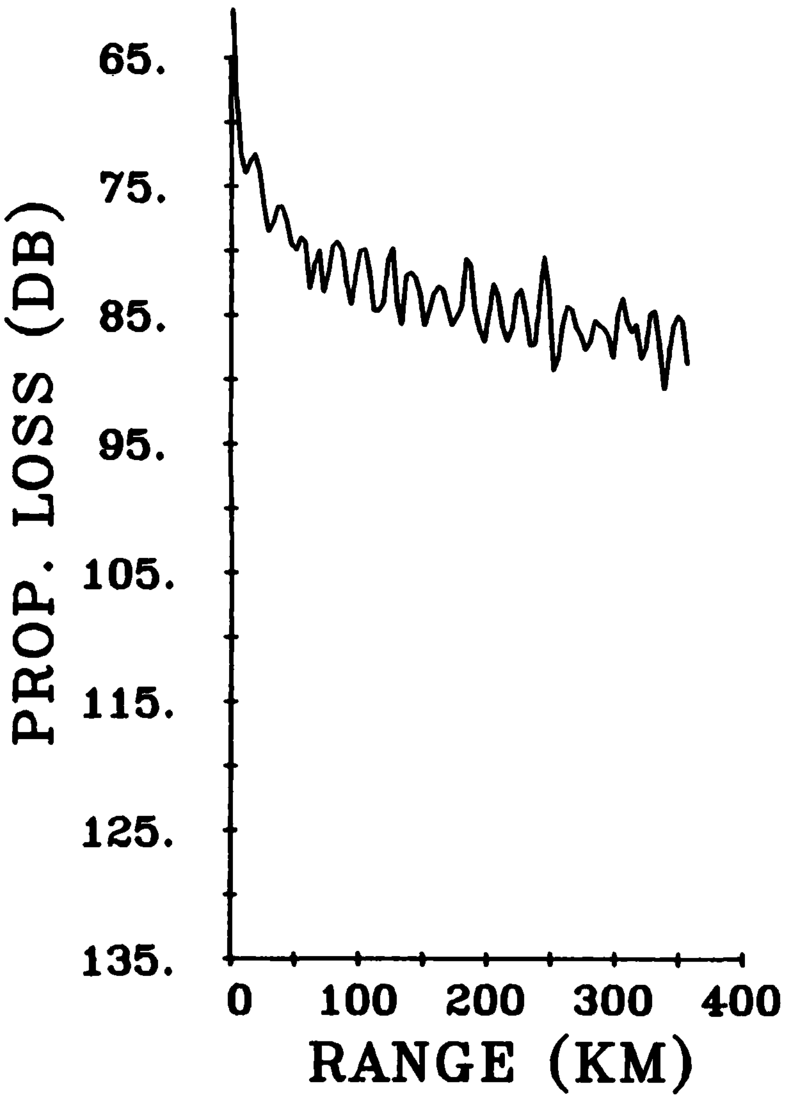


2 50.0 HZ L17H SM 0

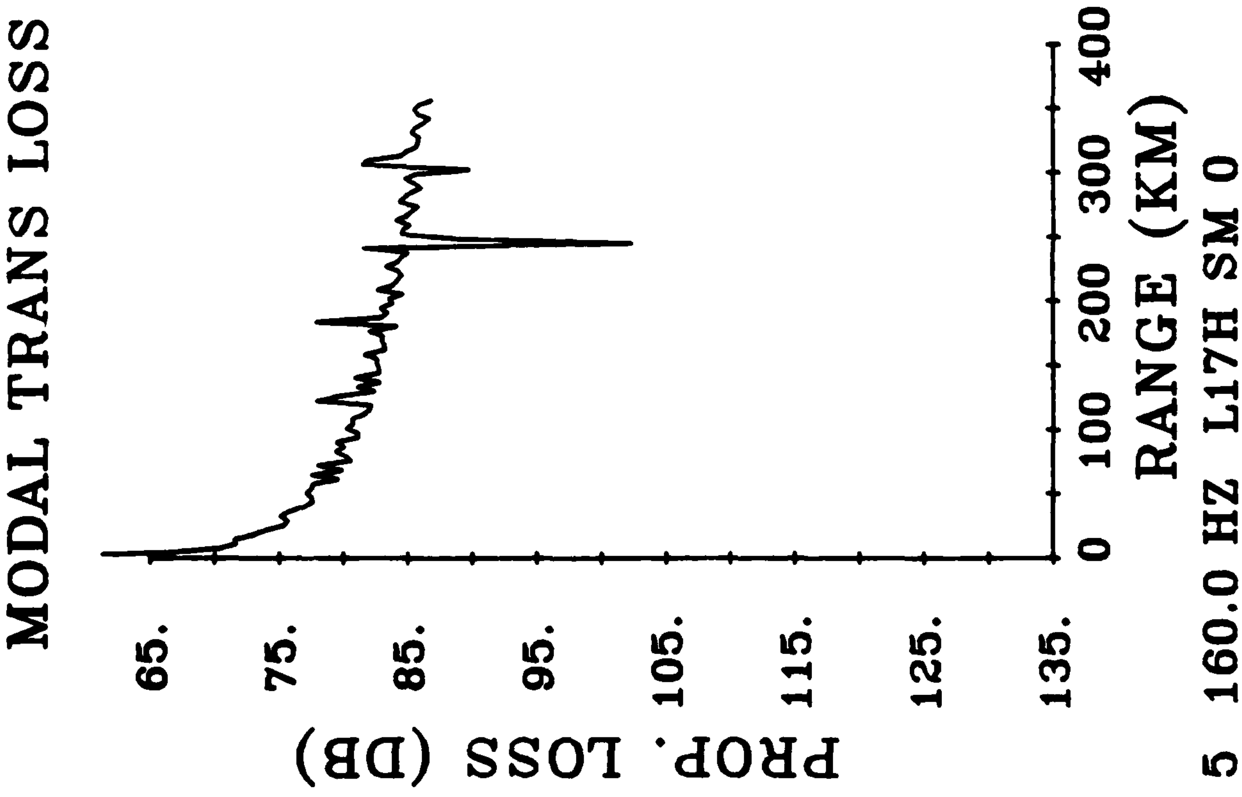
MODAL TRANS LOSS



3 75.0 HZ L17H SM 0

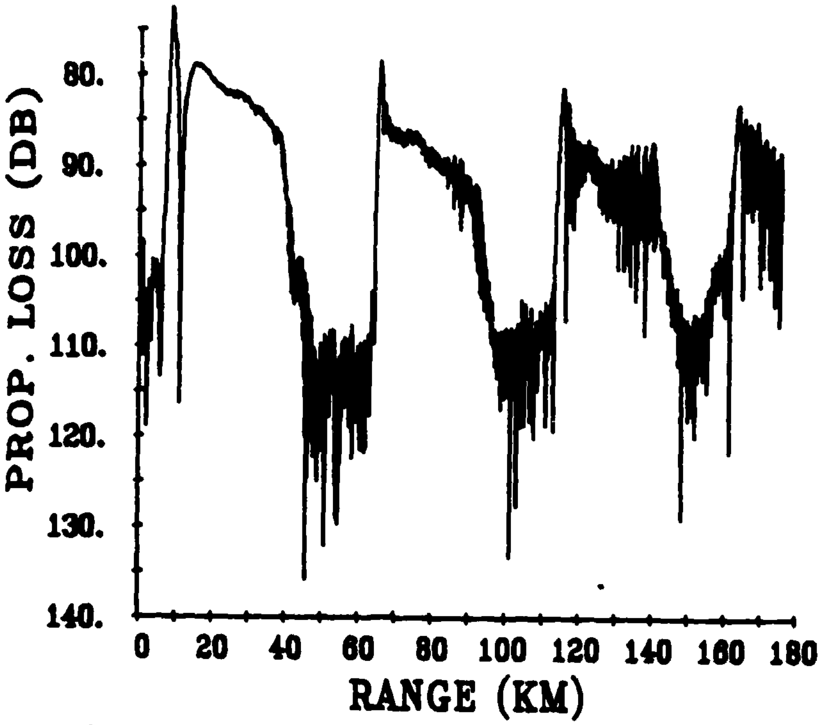


4 120.0 HZ L17H SM 0

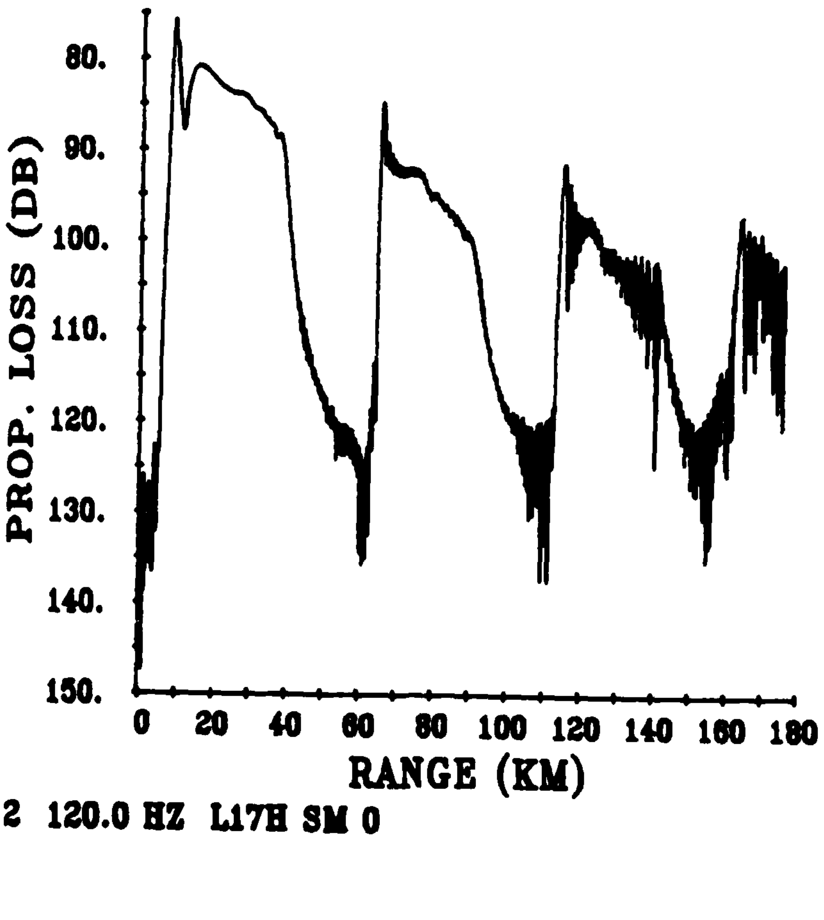

MODAL TRANS LOSS
PROP. LOSS (DB)
65.
75.
85.
95.
105.
115.
125.
135.
0
100
200
300
400
RANGE (KM)
5 160.0 HZ L17H SM 0

```
LSH03AUG.DAT;2                          6-APR-1983 15:30
                       TEST CASE DATA
  7  1  0                                DATA FOR TLOSS17HH
 00.0         1435.0     0.000       0.00001    0.00000    1.03
 50.0         1436.0     0.000       0.00001    0.00000    1.03
 300.0        1457.0     0.000       0.00001    0.00000    1.03
 800.0        1462.0     0.000       0.00001    0.00000    1.03
 3200.0       1500.0     0.000       0.00001    0.00000    1.03
 4400.0       1523.0     0.000       0.00001    0.00000    1.03
 4400.0       1523.0     0.000       0.00001    0.00000    1.03
     2     5       2 2048    01     1      1 120.0      10000.0    1485.0     1600.0
 0.0          00.0       0.00        00.00      0.000            1     0
 0.0          00.0
     2     5       2 2048    01     1      1 120.0      10000.0    1485.0     1600.0
 3.0          00.0       0.00        00.00      0.000            1     0
 0.0          00.0
     2     5       2 2048    00     1      1 120.0      10000.0    1485.0     1600.0
 5.0          00.0       0.00        00.00      0.000            1     0
 0.0          00.0
```

TRANSMISSION LS.

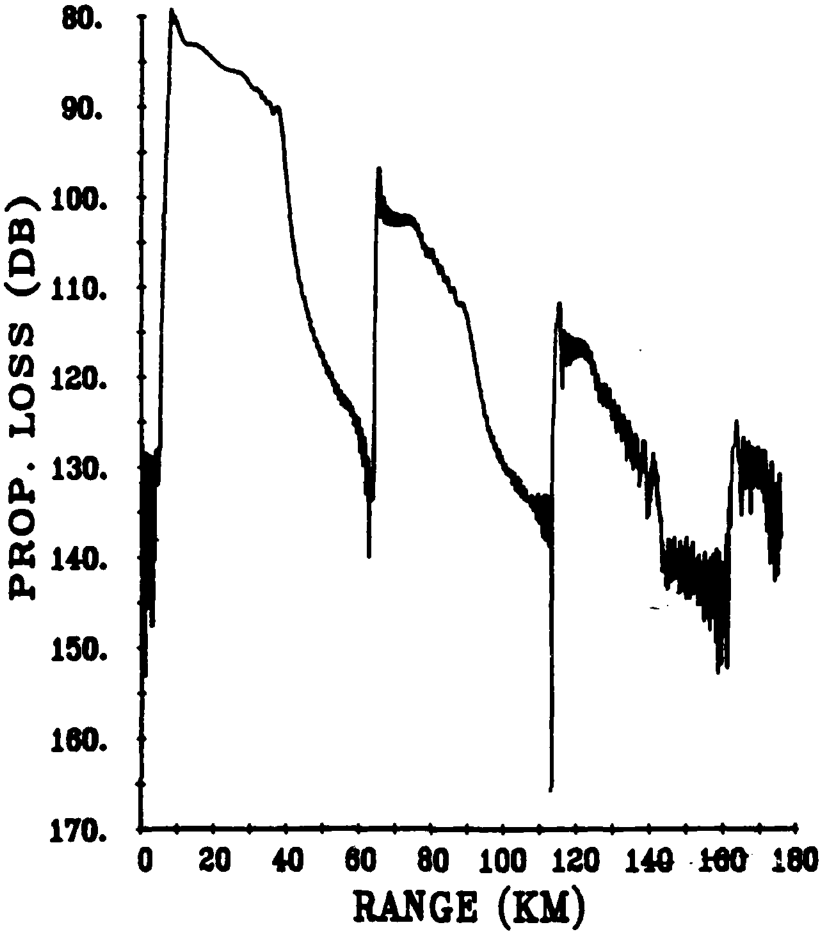


1 120.0 HZ L17H SM 0



2 120.0 HZ L17H SM 0

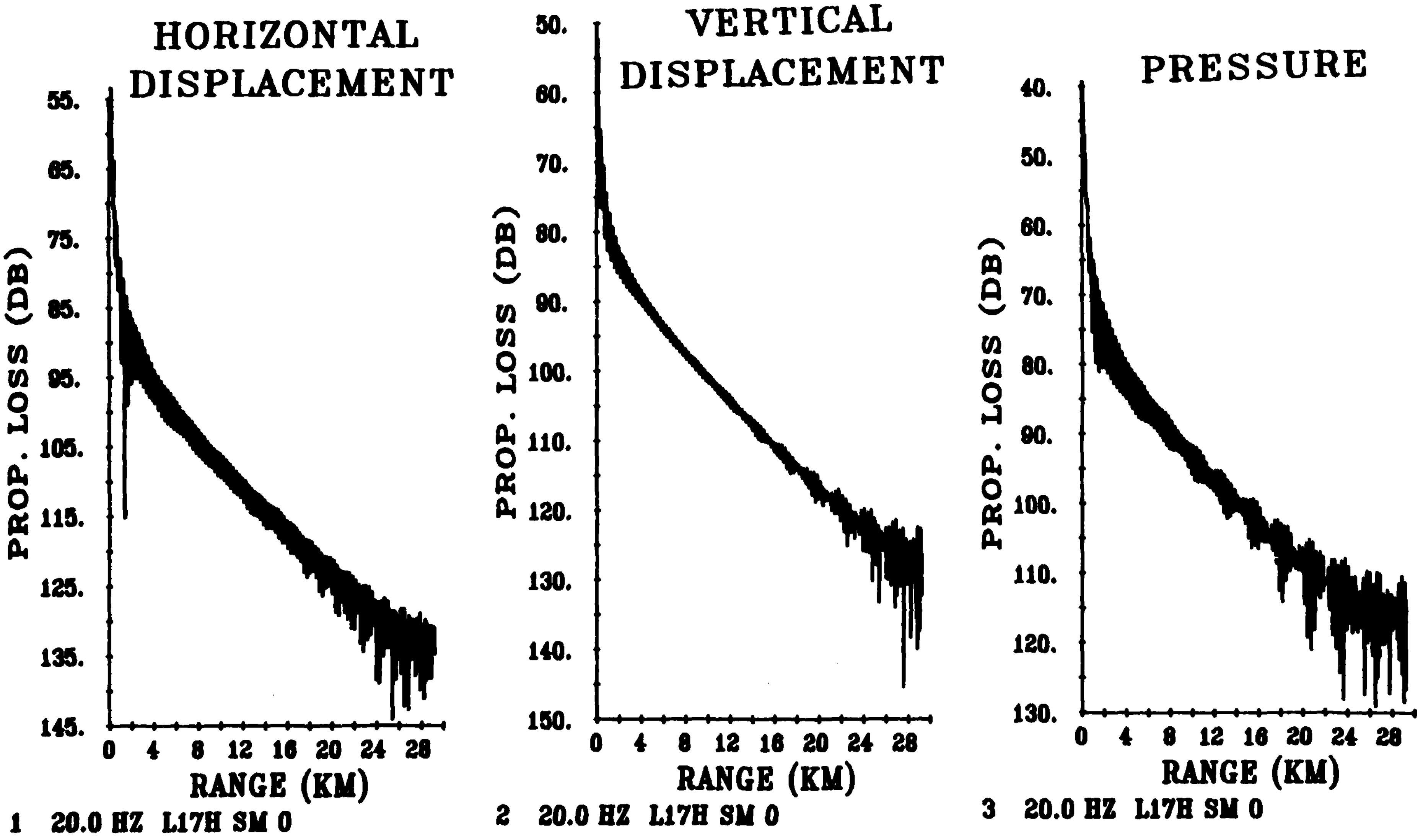
TRANSMISSION LS.



3 120.0 HZ L17H SM 0

```
LSHO3OCT.DAT;2                      6-APR-1983 15:30
                    TEST CASE DATA
06   1                                                     DATA FOR TLOSS17HH
 0.00      1500.0                        .0000150                   1.00
50.00      1500.0                        .0000150                   1.00
100.0      1500.0                        .0000150                   1.00
100.0      3000.0       1400.0           .00007700    .0001600 2.50
500.0      3000.0       1400.0           .00007700    .0001600 2.50
500.0      3000.0       1400.0           .00007700    .0001600 2.50
    2    3    02 1024    -1   000      1    020.0    10000.        1000.0     8000.0
  0.0       1500.0    .0000150    1.00       0000.0            0      1    -1     1
 00.        00.0
06   1                                                     DATA FOR TLOSS17HH
 0.00      1500.0                        .0000150                   1.00
50.00      1500.0                        .0000150                   1.00
100.0      1500.0                        .0000150                   1.00
100.0      3000.0       1400.0           .00007700    .0001600 2.50
500.0      3000.0       1400.0           .00007700    .0001600 2.50
500.0      3000.0       1400.0           .00007700    .0001600 2.50
    2    3    02 1024    -1   000      1    020.0    10000.        1000.0     8000.0
  0.0       1500.0    .0000150    1.00       0000.0            0      1    01     1
 00.        00.0
06   1                                                     DATA FOR TLOSS17HH
 0.00      1500.0                        .0000150                   1.00
50.00      1500.0                        .0000150                   1.00
100.0      1500.0                        .0000150                   1.00
100.0      3000.0       1400.0           .00007700    .0001600 2.50
500.0      3000.0       1400.0           .00007700    .0001600 2.50
500.0      3000.0       1400.0           .00007700    .0001600 2.50
    2    3    02 1024    00   000      1    020.0    10000.        1000.0     8000.0
  0.0       1500.0    .0000150    1.00       0000.0            0      1    00     1
 00.        00.0
```

TRANSMISSION LS.



HORIZONTAL DISPLACEMENT

1 20.0 HZ L17H SM 0

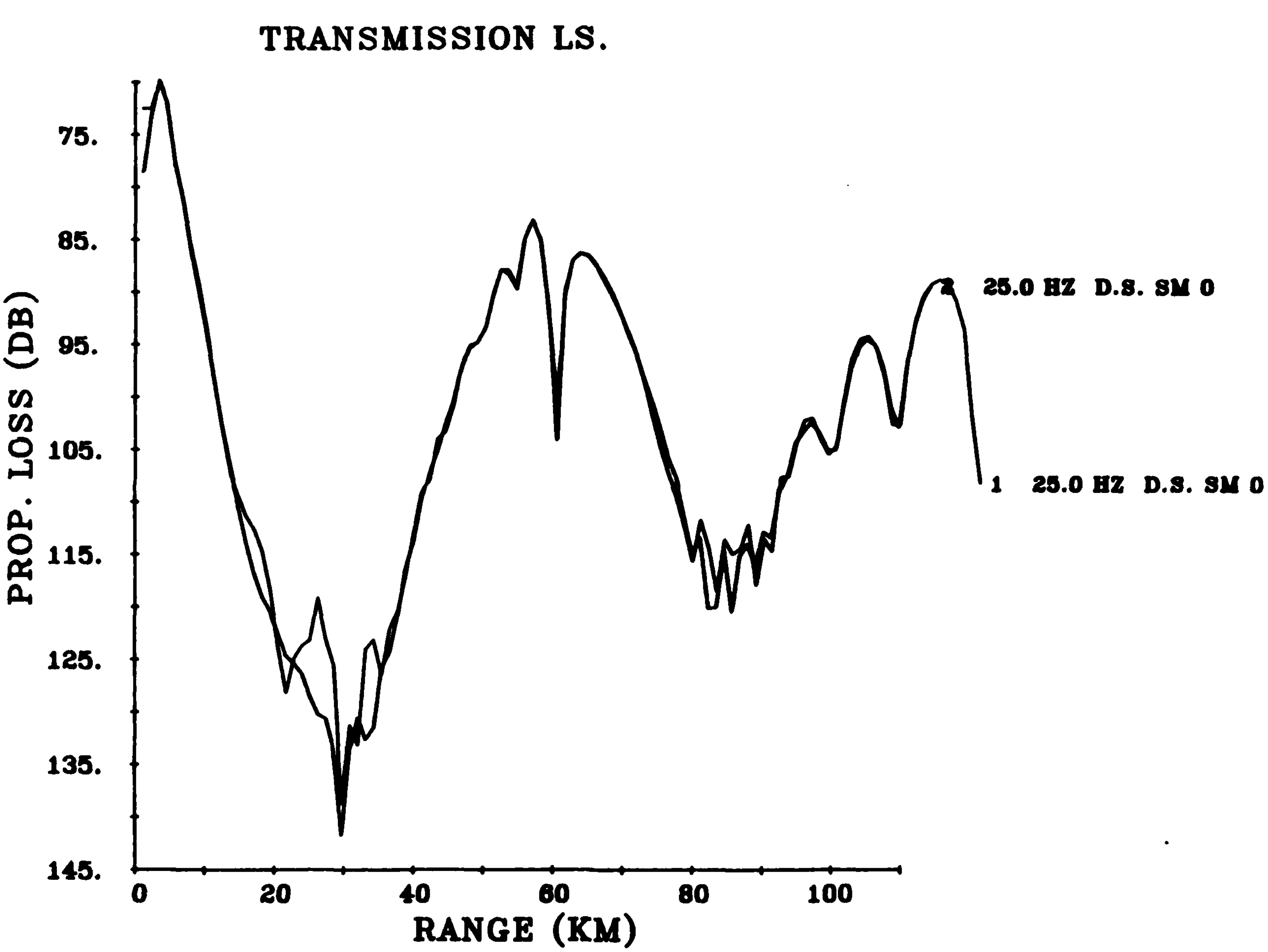
VERTICAL DISPLACEMENT

2 20.0 HZ L17H SM 0

PRESSURE

3 20.0 HZ L17H SM 0

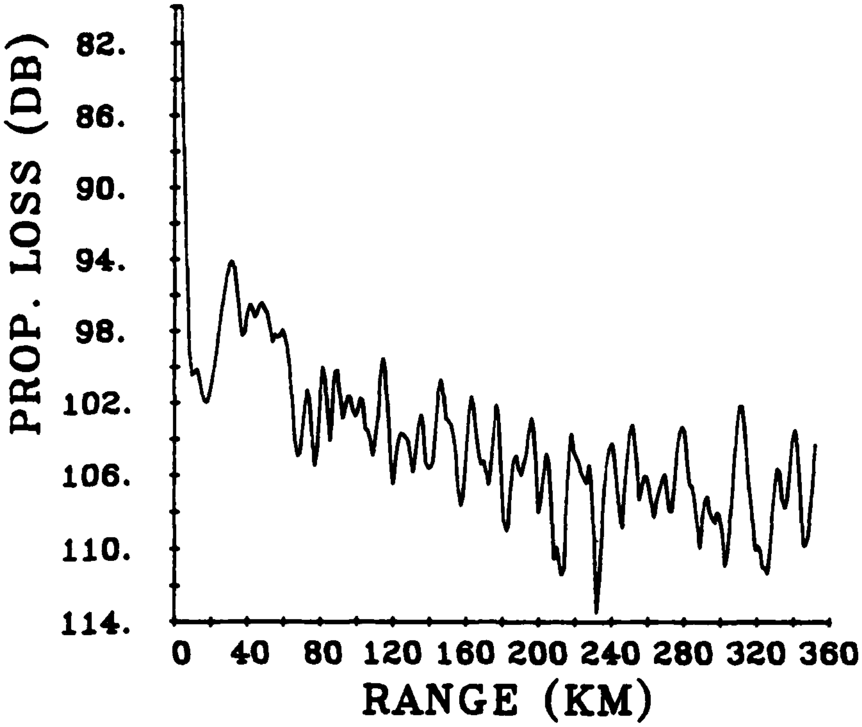
```
LSH12OCT.DAT;2                           6-APR-1983 15:30
                       TEST CASE DATA
  08   1   0                            DATA FOR TLOSS17HH
  000.0        1536.5                   .0000100             1.0
  152.4        1539.2                   .0000100             1.0
  253.9        1530.0                   .0000100             1.0
  406.3        1501.1                   .0000100             1.0
  863.5        1475.0                   .0000100             1.0
  1015.9       1471.9                   .0000100             1.0
  5587.9       1549.6                   .0000100             1.0
  5587.9       1549.6                   .0000100             1.9
     3     5        2  512     -1    00     1    25.     10000.0     1470.0     1549.4
   00.0         0000.0     0.00000      0.0        000.0        0    0
   00.0         0000.0
  08   1   0
  000.0        1536.5                   .0000000             1.0
  152.4        1539.2                   .0000000             1.0
  253.9        1530.0                   .0000000             1.0
  406.3        1501.1                   .0000000             1.0
  863.5        1475.0                   .0000000             1.0
  1015.9       1471.9                   .0000000             1.0
  5587.9       1549.6                   .0000000             1.0
  5587.9       1549.6                   .0000000             1.9
     3     5        2  256     00    00     1    25.     10000.0     1470.0     1549.4
   00.0         0000.0     0.00000      0.0        000.0        0    0
   00.0         0000.0
```


TRANSMISSION LS.
PROP. LOSS (DB)
75.
85.
95.
105.
115.
125.
135.
145.
0
20
40
60
80
100
RANGE (KM)
2 25.0 HZ D.S. SM 0
1 25.0 HZ D.S. SM 0

TEST CASE DATA

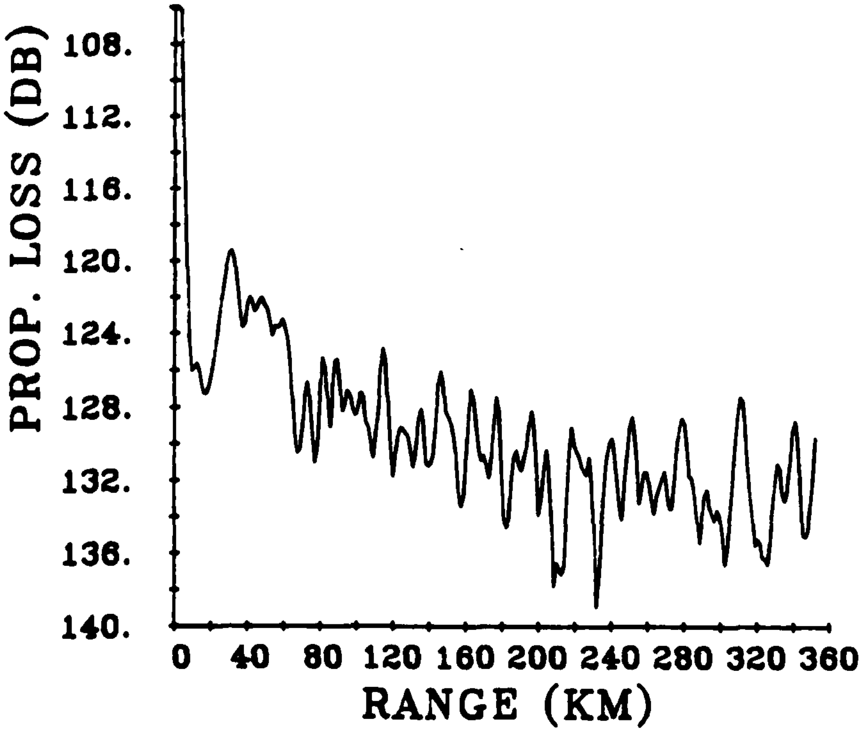
```
10  1  0                       DATA FOR TLOSS17HH
00.0      3800.0    1800.     0.00001   0.00000   0.90
00.0      3800.0    1800.     0.00001   0.00000   0.90
03.0      3800.0    1800.     0.00001   0.00000   0.90
03.0      1435.0    0.000     0.00001   0.00000   1.03
53.0      1436.0    0.000     0.00001   0.00000   1.03
303.0     1457.0    0.000     0.00001   0.00000   1.03
803.0     1462.0    0.000     0.00001   0.00000   1.03
3203.0    1503.0    0.000     0.00001   0.00000   1.03
4403.0    1523.0    0.000     0.00001   0.00000   1.03
4403.0    1523.0    0.000     0.00001   0.00000   1.03
   5    2    2 2048   01     1    1 010.0     10000.0   1434.0    1600.0
0.0       00.0      0.00      00.00     0.000        1    0   01
0.0       00.0
   5    2    2 2048   00     1    1 010.0     10000.0   1434.0    1600.0
0.0       00.0      0.00      00.00     0.000        1    0   -1
0.0       00.0
```

TRANSMISSION LS.



1 10.0 HZ L17H SM 2

TRANSMISSION LS.



2 10.0 HZ L17H SM 2

TLOSS51HH USER'S MANUAL

A. Introduction

TLOSS51HH, our most complete transmission loss (hereafter abbreviated as TL) program, was completed in February of 1983. It is an extension of TLOSS17HH, which is an extension of TLOSS10HH, the interim standard TL program.
TLOSS51HH has these features and options for the user:

1. A choice of computational methods;
   a. The FFP method
   b. The Normal Mode method

   If the Normal Mode method is chosen, the user also has the option of including either of two branch cut integrals.

2. Optional horizontal or vertical displacement in addition to the usual pressure values for TL.

3. A comprehensive selection of modelling parameters. These include:
   a. A detailed sound velocity profile
   b. Choice of source depth
   c. Choice of detector depth
   d. Choice of frequency
   e. Choice of wave number integrand limits
   f. Optional beaming, as it is affected by integrand tapering (in the FFP mode)
   g. Optional surface roughness
   h. Choice of either fluid or vacuum upper half space
   i. Possibility of the source and detector in the fluid upper half space
   j. Optional internal roughness
   k. Choice of initial range, range step, and range count (for the Normal Mode method)

4. Output. This includes:
   a. A printout of all input
   b. The TL plot
   c. The wave integrand plot of the FFP; either short or long.
   d. A table of the denominator values for the Normal Mode method
   e. A table of roots and their corresponding phase velocities, slopes, and excitation functions for the Normal Mode method

   Note: All short plots except the wave integrands are repeated, the repeated plot being a smoothed copy of the first and having the same bounds as the first.

5. Additional Optional Tables and Plots. These include:
   a. The TL plot in nautical miles, if desired and if the range is sufficiently long. The original TL plot is in kilometers.
   b. The denominator plot for the Normal Mode method.
   c. Modal TL plots, Branch TL plots, and Total TL plots, if a branch cut option is chosen.
   d. A long printer-plot of the sound velocity profile after the profile has been broken into equal increment depths and also a table of this expanded sound velocity profile.
   e. Eigen value printer-plots for each mode, if there are mode

6. Numerical output of the TL for further processing or plotting. Also, eigen values are put out in a separate file, if there are an Formerly, this was called punched output, since it went out to the card punch.

7. A profile check for multiple ducts. This check is made for the Normal Mode method. If multiple ducts are present, special care is taken to locate those 'hidden' roots which are only found by reducing the step size of the denominator function.

B. Input Variables and Formats

Program input for TLOSS51HH includes both the physical model and program control. This section of the manual will list and explain each line of the input.

1. Entry 1, one line

```
MMAXF, NCARD                                  variable names
( 2I3 )                                       format
```

MMAXF is a control variable which is used to read in the correct number of points in the sound velocity profile. (see 2. - next Entry)

NCARD is a control variable that, when set non-zero allows the program to put out TL data on unit 7 in a form which is readable as input by another program. A zero value will prevent any output on unit 7. This is normally used to make graphic plots. (Originally, on the IBM 360/91, unit 7 was the punched card unit, hence the name NCARD.)

2. Entry 2, MMAXF lines (max. of 49 lines)

```
(( V(M,N), N=1,7 ), M=1,MMAXF )               variable name
( 7D10.0 )                                    format
```

V is an array that contains the sound velocity profile. For each point in the profile, and on each line, (one point per line), the program expects:

a. a depth value, in meters
b. a sound (pressure) velocity value, in meters per second
c. a sound (shear) velocity value, in meters per second, if applicable
d. a corresponding attenuation value for pressure
e. a corresponding attenuation value for shear, if applicable
f. a density value, in grams per c.c., or specific density
g. a roughness value, if applicable

Note: No depth value can be less than a previous depth value. The program will stop if this is so.

3. Entry 3, one line

```
KSF, LRF, MM, LS, KKK, IBEAM, IMAX, FREQ, PO, C, CMAX
                                              variable names
( 7I5, 4D10.0 )                               format
```

KSF is a pointer to the source depth value. It points to one of the points in the sound velocity profile; the one which is the source depth value.

LRF is a pointer to the detector (or receiver) depth value. It's function is similar to KSF.

MM is a TL plot pruning factor, with a minimum acceptable value of 2. MM is applicable to FFP only. If an MM value of N is read in, then the first 1/N of the plot will be plotted. This is useful either to get a more detailed look at the output or to prune away that part of the plot affected by range aliasing.

LS is the number of points used in the wave integration. It also affects the range of the TL plot, for the FFP method.

KKK is a control variable that indicates:

a. If zero, this is the last case.
b. If negative, the next case starts with Entry 1, (new profile).
c. If positive, the next case starts with Entry 3.

IBEAM is a control variable that, when set non-zero, requires the program to taper the wave integrand at each end.

IMAX is the range count or the number of points in the TL plot for a Normal Mode case.

FREQ is the frequency for this case, in Hertz.

PO is the source level. For TL PO = 1.0

C is the lower bound sound velocity in the wave integration, in meters per second. For the Normal Mode method, if a zero is entered, a default value will be found from the sound velocity profile.

CMAX is the upper bound sound velocity in the wave integration, in meters per second. If a zero is entered here, the wave number integration will start from zero for the FFP method, or, for the Normal Mode method, a default value will be found from the sound velocity profile.

4. Entry 4, one line

```
REFLTO, ALPHAO, ATTENO, RHOO, H, INAUT, LONG, KHPV, IBR,
IDPL                                      variable names
( 5D10.0, 5I5 )                           format
```

REFLTO is the RMS surface roughness value, in meters. If this value is non-zero, the program explicitly sets the next three values to zero.

ALPHAO is, if present, the sound velocity in the fluid upper half space, in meters per second. This is the variable that determines whether the present case has either a fluid or vacuum upper half space.

ATTENO is, if present, the attenuation in the fluid upper half space.

RHOO is, if present, the density of the fluid upper half space, in grams per c.c.

H is, if present, the distance between the source and detector in the fluid upper half space. To use this option:

a. The first two sound velocity depths must be zero.
b. There must be some variation in the pressure sound velocity profile.
c. There must not be any shear sound velocity.
d. ALPHAO and H must be present.

INAUT is a control variable that, when set non-zero, allows the plotting of the TL plot in nautical miles if the range is greater than 100 kilometers. This is for the FFP method only.

LONG is a control variable that, when set non-zero, switches the wave integrand plot from short to long. This is used when it is necessary to examine each point of the wave integrand. FFP only.

KHPV is the control variable that determines whether output will be:

a. pressure, if zero.
b. vertical displacement, if positive.
c. horizontal displacement, if negative.

IBR is the control variable that determines if a branch cut integral will be included with the Normal Mode method.

a. If zero, no.
b. If positive, yes (exact)
c. If negative, yes (FFP)

IDPL is the control variable that, when set, allows the denominator function to be plotted. Normal Mode method only.

5. Entry 5, one line

```
RA, DLRA, THLAYR, IEXP                 variable names
( 3D10.0, I5 )                         format
```

RA is the initial range of the TL plot, in meters.

DLRA is the range step of the TL plot, in meters.

These two variables are read in for the Normal Mode method, when IBR is zero or positive, only. Since the TL plot is in kilometers, it is very important to be careful with the units of these two variables.

THLAYR is the desired layer thickness for the sound velocity profile which is used if IEXP is set non-zero. New interface depths are chosen by interpolation every THLAYR meters and the original data points are retained. This is valid for FFP and Modes.

IEXP is a control flag which, when set non-zero, directs the program to split up the input sound velocity profile and calculate TL on this new profile. If this is a normal mode run, and if there are modes then plot out on the printer eigen value plots for each mode. Also, eigen values vs. depth are put out on Unit 8 to a file for further processing or plotting.

Note:

The program determines the computational method, (whether FFP or Normal Mode), by examining all of the attenuation values and the surface roughness value. If they are all zero, the Normal Mode method is selected.

C. Operating Procedures

The primary computer at Lamont-Doherty Geological Observatory is a VAX 11/780, which runs under the VMS operating system. Listed below are four distinctly different operations that are used to process TLOSS51HH.

1. Compiling-Linking

a. Compiling (FORT command)

Compiling the TLOSS51HH program on the VAX 11/780 has always been done with the following switches on:

1. /CHECK Forbids array pointer overflow.
2. /NOI4 Integer variables are set to 2 bytes.
3. /NOF77 Where the compiler needs to choose between Fortran 77 and Fortran 4+, the latter is chosen.
4. /LIST A compiler listing is made and stored on the disk as TLOSS51HH.LIS.

b. Linking (LINK command)

Linking is straight-forward. No extra routines are linked with TLOSS51HH.

2. Running

TLOSS51HH is normally run through a batch stream, in particular the SLOW batch stream, since this is the cheapest route, especially at night. However, there is another batch stream with higher priority (and higher cost) that can be used for quicker results, the SYS$BATCH batch stream. In order to use either of these two batch streams:

a. The executable file (TLOSS51HH.EXE) must be ready to use. The LINK operation satisfies this requirement.

b. The data file must also be ready to use. This is normally prepared by the user using the editor, (and the TLOSS51HH USER'S MANUAL, if necessary) at a terminal.

c. Submit the command file, TLOSS51HH.COM, to the batch stream (or batch queue) using a command such as one of these:

```
SUBMIT/NAME=LF022FEB/QUEUE=SLOW/AFTER=20:00 TLOSS51HH
      SLOW queue, Job starts at 8:00 P.M.

SUBMIT/NAME=LF022FEB/QUEUE=SLOW TLOSS51HH
      SLOW queue, Job starts now

SUBMIT/NAME=LF022FEB TLOSS51HH
      SYS$BATCH queue ( the 'regular' queue ),
      Job starts now
```

d. The following command file, also prepared using the editor, is a typical file submitted, using one of the previous submit commands.

```
$ASSIGN LF022FEB SYS$PRINT                   (1)
$ASSIGN LF022FEB SYS$OUTPUT                  (2)
$ASSIGN DR0:LF022FEB.PLT FOR007              (3)
$ASSIGN DR0:LF022FEB.OUT FOR006              (4)
$ASSIGN DR0:SIMPLE19.DAF FOR005              (5)
$TYPE TLOSS51HH                              (6)
$COPY DR0:SIMPLE19.DAF DR0:LF022FEB.DAT      (7)
$RUN TLOSS51HH                               (8)
```

This run is called the LF022FEB run. 'LFO' stands for LOSS FIFTY ONE H. '22FEB' indicates that this is the 22nd run in February.

The first two lines in this command file put the log file on the disk instead of the printer.
The third line puts unit 7 output into file LFO22FEB.PLT.
The fourth line puts unit 6 output into file LFO22FEB.OUT.
The fifth line draws unit 5 input from SIMPLE19.DAF.
The sixth line copies TLOSS51HH.LIS to LFO22FEB.LIS.
The seventh line copies SIMPLE19.DAF to LFO22FEB.DAT.
The eighth line runs program TLOSS51HH.

A command file such as this yields a file group; each file having the same name but a different extension. These files are then treated collectively. They are printed together and stored on tape together. Such a group is, for example:

LFO22FEB.DAT	a copy of the data file
LFO22FEB.LIS	the program listing file
LFO22FEB.LOG	the log file
LFO22FEB.OUT	the output file
LFO22FEB.PLT	the numerical output file

The user normally gets a complete printout of the run by typing the command:

```
PRINT LFO22FEB.*;*
```

on the terminal.

3. Copying Runs to Tape (Archiving)

Periodically, it is necessary to clear the old runs off the disk and put them on tape. This is a very simple operation and proceeds as follows:

a. Initialize a tape. Give it a name that is somehow indicative of the present month. For example:

```
INITIALIZE/DENSITY=1600 MT1: FEBMAR
```

This tape would have runs made in February and March. This step is not necessary if the user is adding runs to a tape that is already partly filled. In such a case, be careful not to initialize!

b. Mount the tape with the MOUNT command.

```
MOUNT MT1: FEBMAR
```

This is not the same as physically mounting the tape on the tape drive, which is assumed to have taken place previously.

c. Copy all the run files from disk to tape with the COPY command. For example in this case:

```
COPY *FEB.*;* MT1:
COPY *MAR.*;* MT1:
```

would copy all the February and all the March runs onto the tape mounted on tape drive 1.

d. For future reference, it is good to get a list of names of the files on the tape. This is accomplished as follows:

```
DIRECTORY/PRINT MT1:
```

The printout resulting from this command does not have the tape name on it, so the user should write this on immediately.
If, for some reason, this printout is lost, the tape can be remounted and the directory can be taken again. But this is no reason to be careless with the original directory printout.

e. Dismount the tape with the dismount command:

```
DISMOUNT MT1:
```

f. Delete the files on the disk that have just been copied to the tape. In this case:

```
DELETE *FEB.*;*
DELETE *MAR.*;*
```

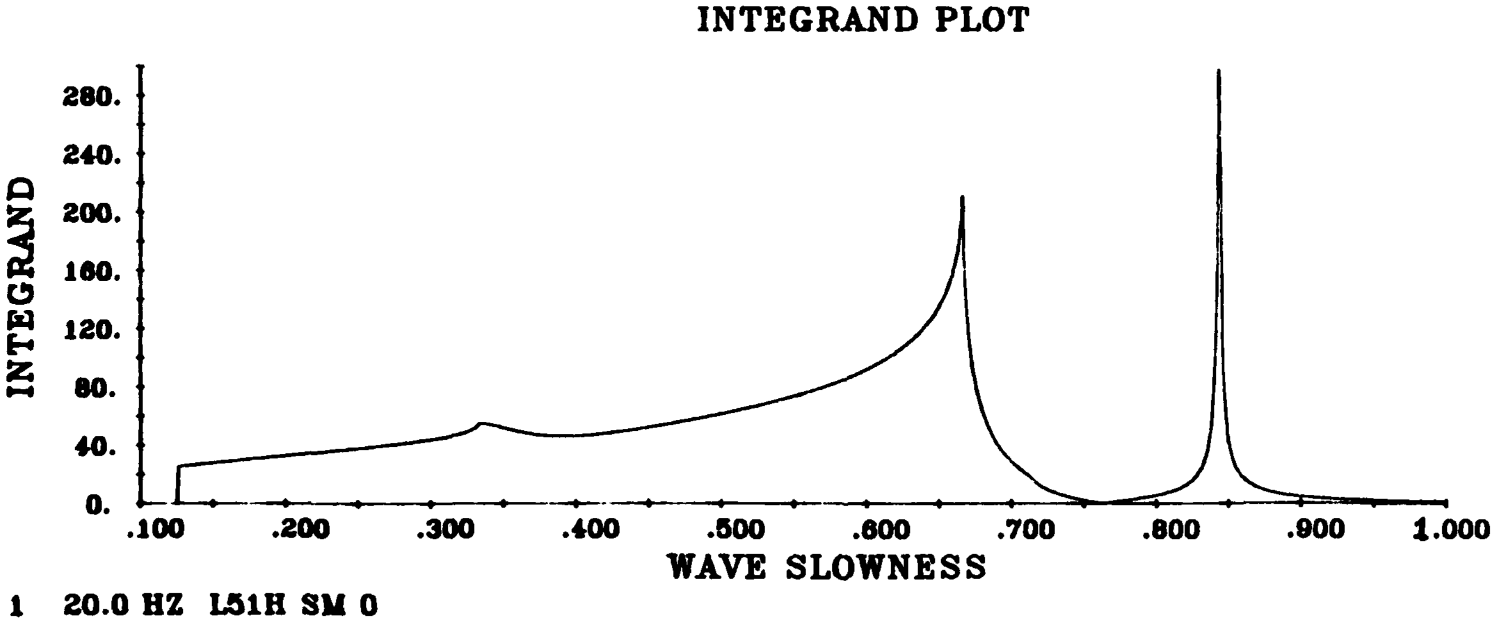
4. Distributing the Program by Tape

The program and test cases are distributed using tapes with the same format as the archival tapes. This is the Files-11 format, peculiar to Digital Equipment Corporation. The tapes from this system have a density of 1600 bpi, and a blocksize of 2048 bytes.

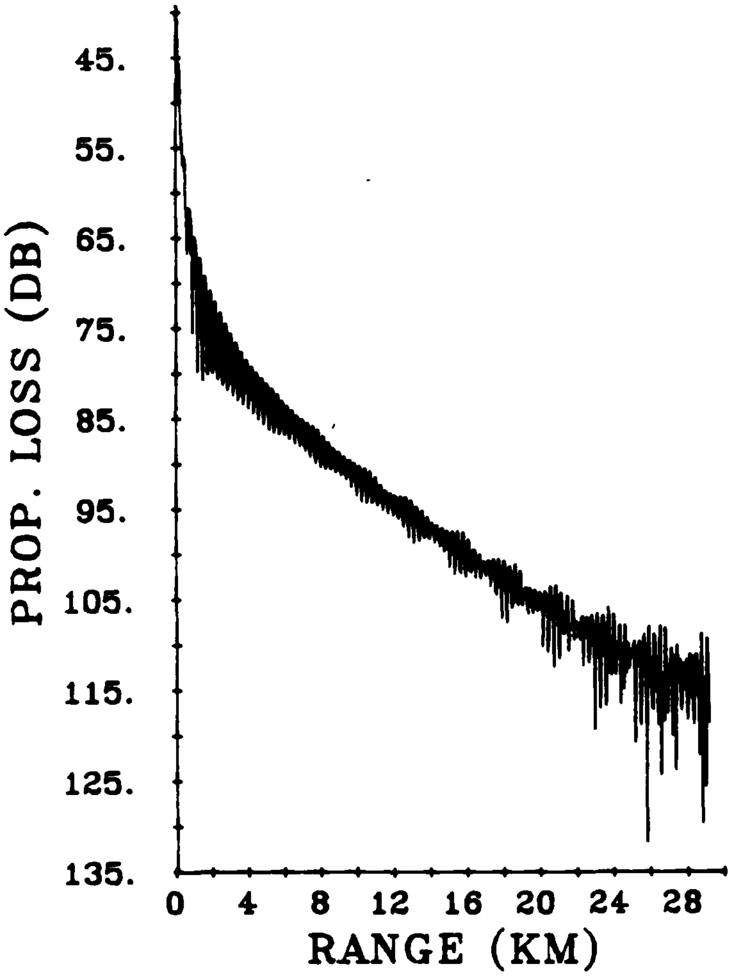
If any further questions arise, feel free to call:

Dr. H. Kutschale	(914) 359-2900 x382
Mr. L. Carroll	(914) 359-2900 x221

```
LF002MAR.DAT;2                   6-APR-1983 15:30
                TEST CASE DATA
 06  1                                   DATA FOR TLOSS51HH
  0.00     1500.0              .0000000            1.00
 50.00     1500.0              .0000000            1.00
 100.0     1500.0              .0000000            1.00
 100.0     3000.0    1400.0    .00006286  .0001281 2.50
 125.0     3000.0    1400.0    .00006286  .0001281 2.50
 125.0     3000.0    1400.0    .00006286  .0001281 2.50
    2    3   02 1024   00  000    1   020.0   10000.    1000.0    8000.0
 0.0        1500.0    .0000000  1.00     0000.0       0    0   00    0
  00.       00.0     00.000       1
```


INTEGRAND PLOT
INTEGRAND
280.
240.
200.
160.
120.
80.
40.
0.
.100
.200
.300
.400
.500
.600
.700
.800
.900
1.000
WAVE SLOWNESS
1 20.0 HZ L51H SM 0

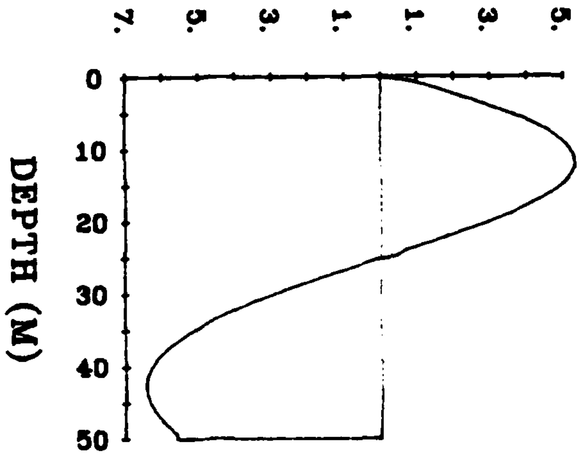
TRANSMISSION LS.



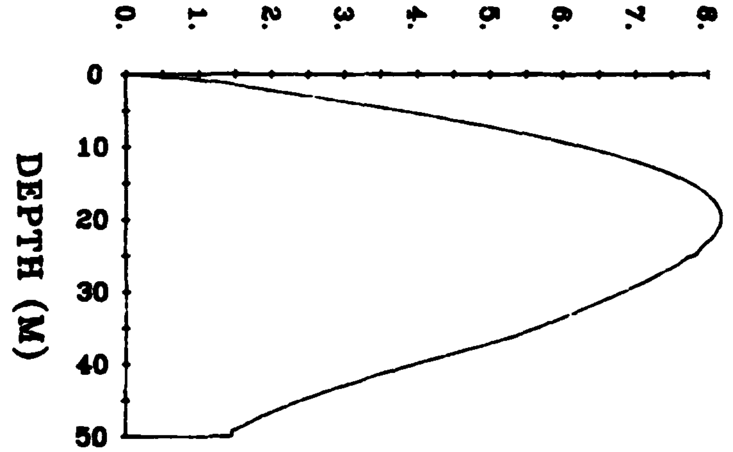
1 20.0 HZ L51H SM 0

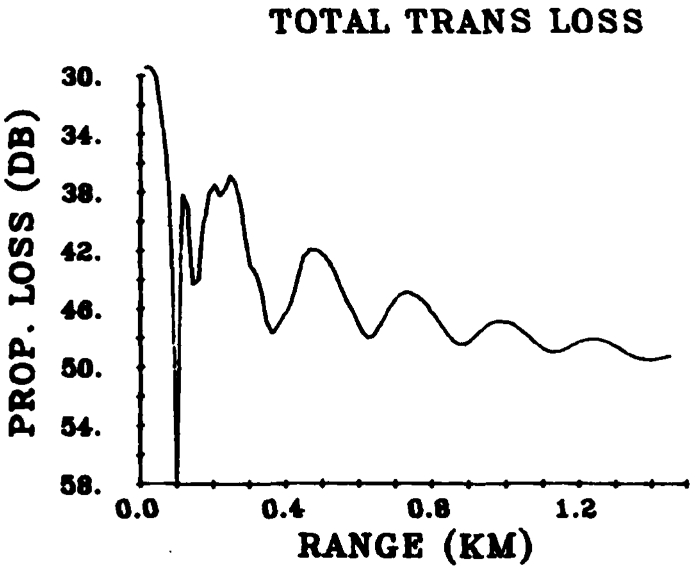
```
LF003MAR.DAT;3                   6-APR-1983 15:30
               TEST CASE DATA

04  1  0                        DATA FOR TLOSS51HH
00.0      1500.0    0.000     0.00000   0.00000  1.00
25.0      1525.0    0.000     0.00000   0.00000  1.00
50.0      1550.0    0.000     0.00000   0.00000  1.00
50.0      1600.0    0.000     0.00000   0.00000  2.00
   2    2    2 1024   00     0  100 110.0     10000.0   1500.001  1599.999
0.0       00.0      0.00      00.00     0.000        0    0    0   -1
0200.0    200.0     0.00         1     RANGE & DLRANGE IN METERS, OUTPUT IN
```

EIGEN PLOT
PRESSURE

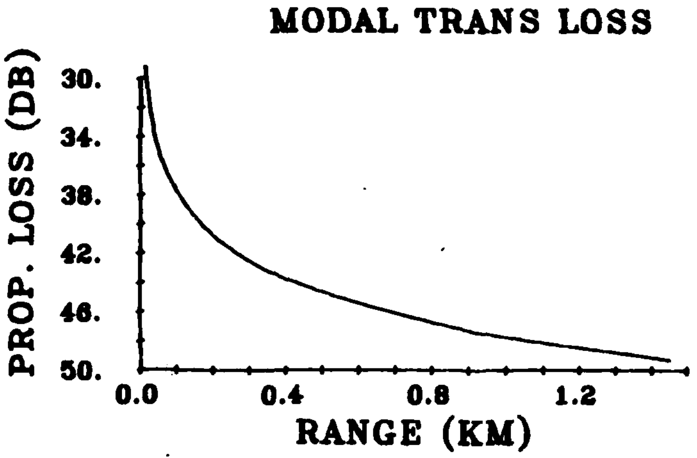


EIGEN PLOT
PRESSURE

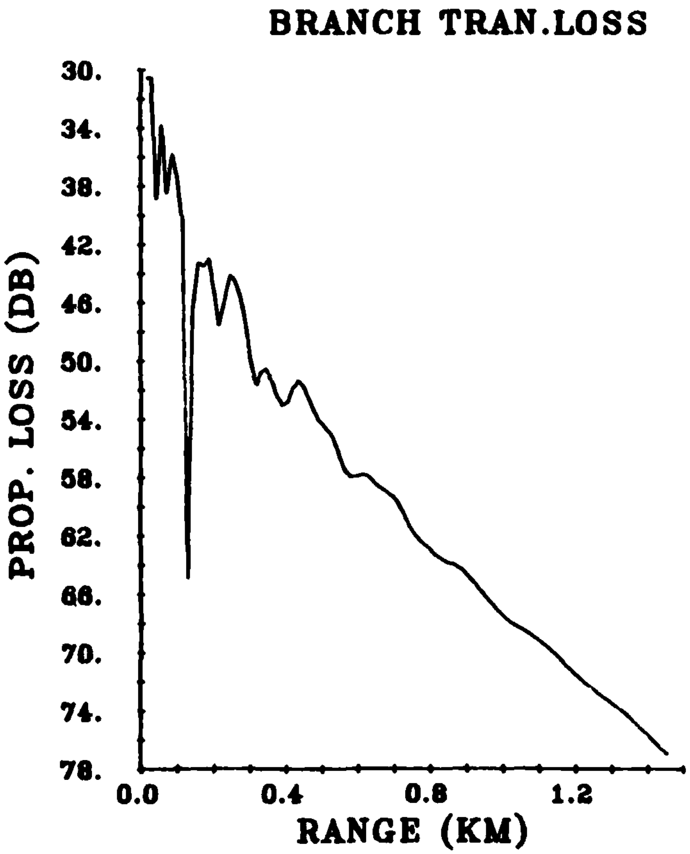




2 110.0 HZ L51H SM 0

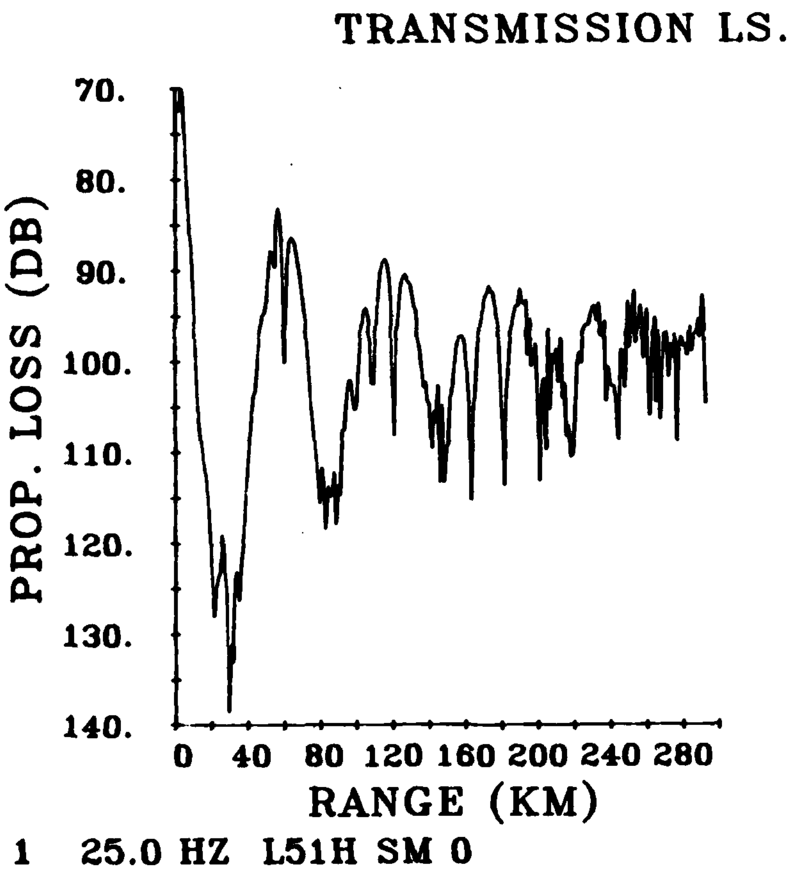


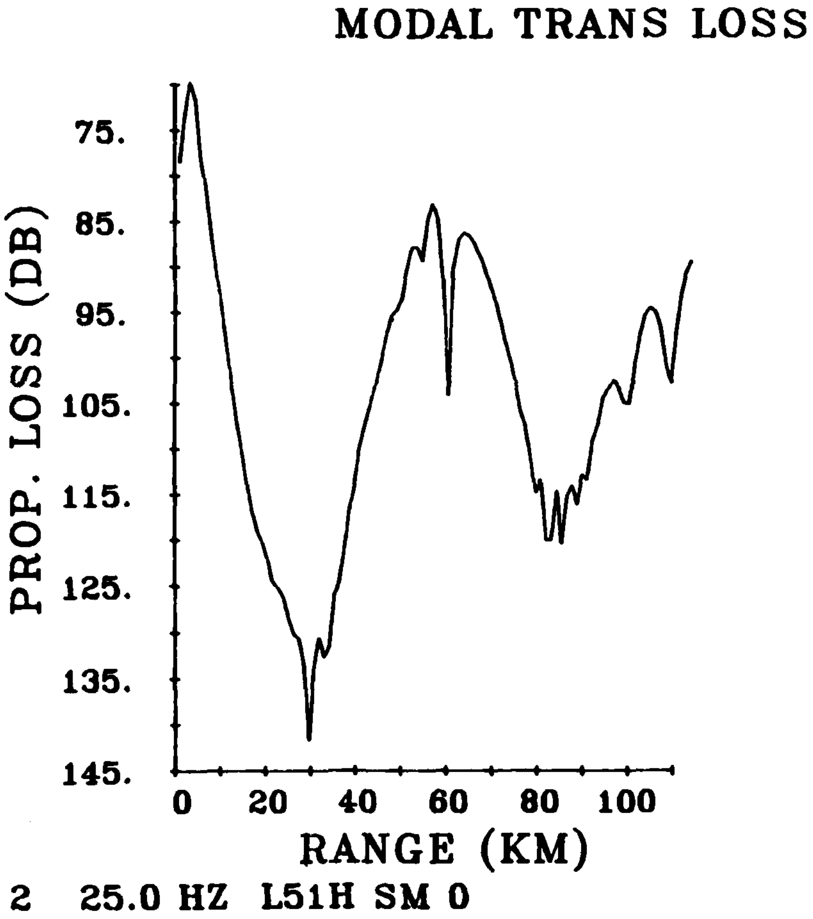
1 110.0 HZ L51H SM 0



3 110.0 HZ L51H SM 0

```
                    TEST CASE DATA
 08   1   0
 000.0       1536.5                    .0000100              1.0
 152.4       1539.2                    .0000100              1.0
 253.9       1530.0                    .0000100              1.0
 406.3       1501.1                    .0000100              1.0
 863.5       1475.0                    .0000100              1.0
 1015.9      1471.9                    .0000100              1.0
 5587.9      1549.6                    .0000100              1.0
 5587.9      1549.6                    .0000100              1.9
     3     5      2  512    -1    00      1    25.      10000.0     1470.0     1549.4
  00.0        0000.0     0.00000     0.0          000.0         0      0
  00.0        0000.0
 08   1   0
 000.0       1536.5                    .0000000              1.0
 152.4       1539.2                    .0000000              1.0
 253.9       1530.0                    .0000000              1.0
 406.3       1501.1                    .0000000              1.0
 863.5       1475.0                    .0000000              1.0
 1015.9      1471.9                    .0000000              1.0
 5587.9      1549.6                    .0000000              1.0
 5587.9      1549.6                    .0000000              1.9
     3     5      2  256    00    00      1    25.      10000.0     1470.0     1549.4
  00.0        0000.0     0.00000     0.0          000.0         0      0
  00.0        0000.0
```

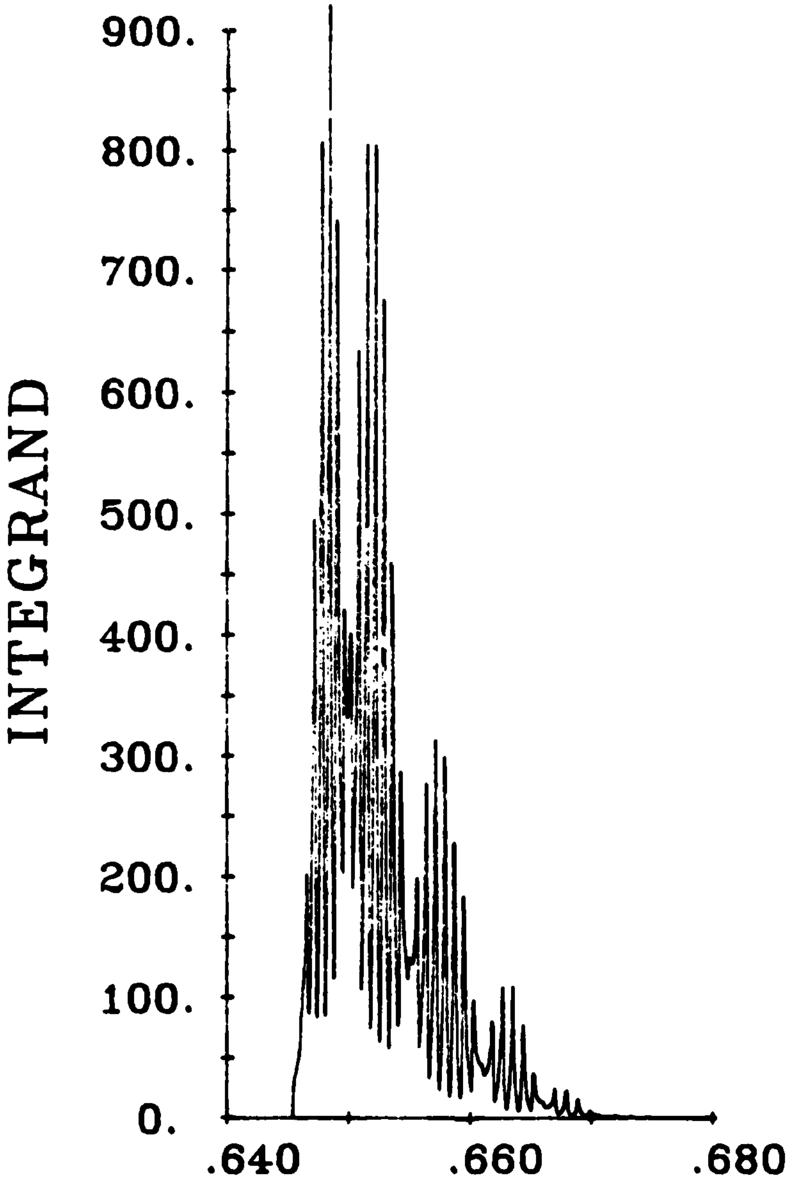

TRANSMISSION LS.
PROP. LOSS (DB)
70.
80.
90.
100.
110.
120.
130.
140.
0
40
80
120
160
200
240
280
RANGE (KM)
1 25.0 HZ L51H SM 0


MODAL TRANS LOSS
PROP. LOSS (DB)
75.
85.
95.
105.
115.
125.
135.
145.
0
20
40
60
80
100
RANGE (KM)
2 25.0 HZ L51H SM 0

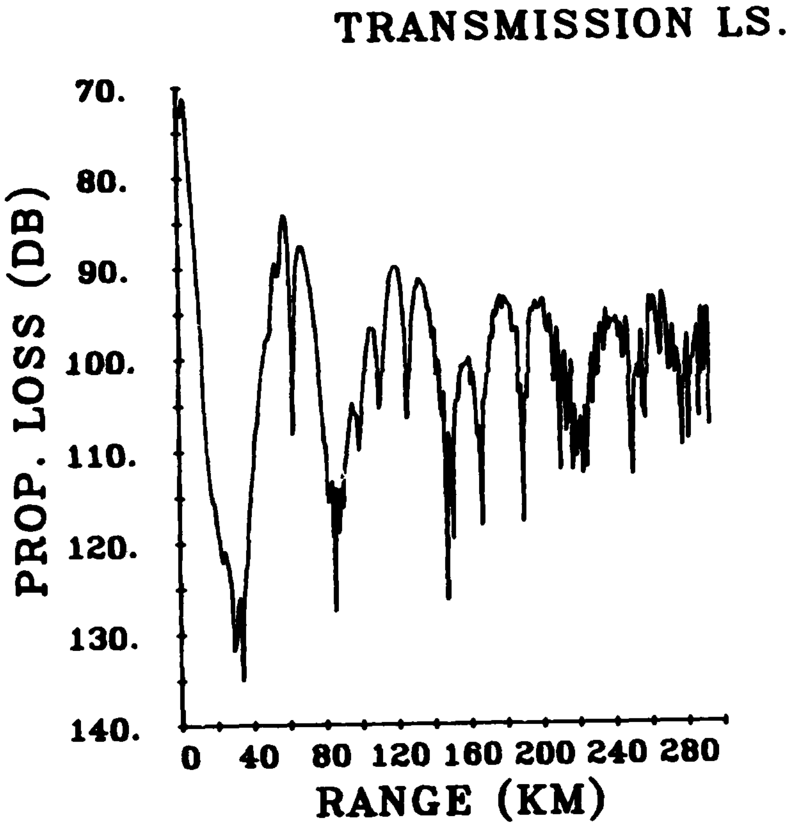
TEST CASE DATA

```
08  1  0
000.0     1536.5              .0000100            1.0
152.4     1539.2              .0000100            1.0
253.9     1530.0              .0000100            1.0
406.3     1501.1              .0000100            1.0
863.5     1475.0              .0000100            1.0
1015.9    1471.9              .0000100            1.0
5587.9    1549.6              .0000100            1.0
5587.9    1549.6              .0000100            1.9
   3    5    2  512  -1     00     1   25.    10000.0    1470.0    1549.4
 00.0      0000.0    0.00000   0.0       000.0       0    0
 00.0      0000.0   050.00        1
08  1  0
000.0     1536.5              .0000000            1.0
152.4     1539.2              .0000000            1.0
253.9     1530.0              .0000000            1.0
406.3     1501.1              .0000000            1.0
863.5     1475.0              .0000000            1.0
1015.9    1471.9              .0000000            1.0
5587.9    1549.6              .0000000            1.0
5587.9    1549.6              .0000000            1.9
   3    5    2  256   00    00     1   25.    10000.0    1470.0    1549.4
 00.0      0000.0    0.00000   0.0       000.0       0    0
 00.0      0000.0   050.00        1
```

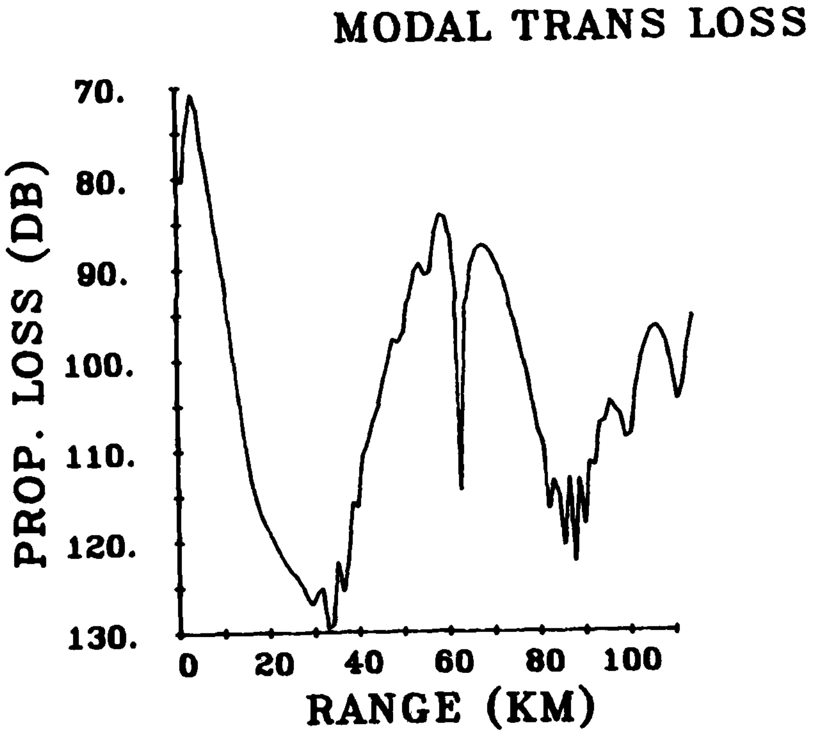
INTEGRAND PLOT



1 25.0 HZ L51H SM 0



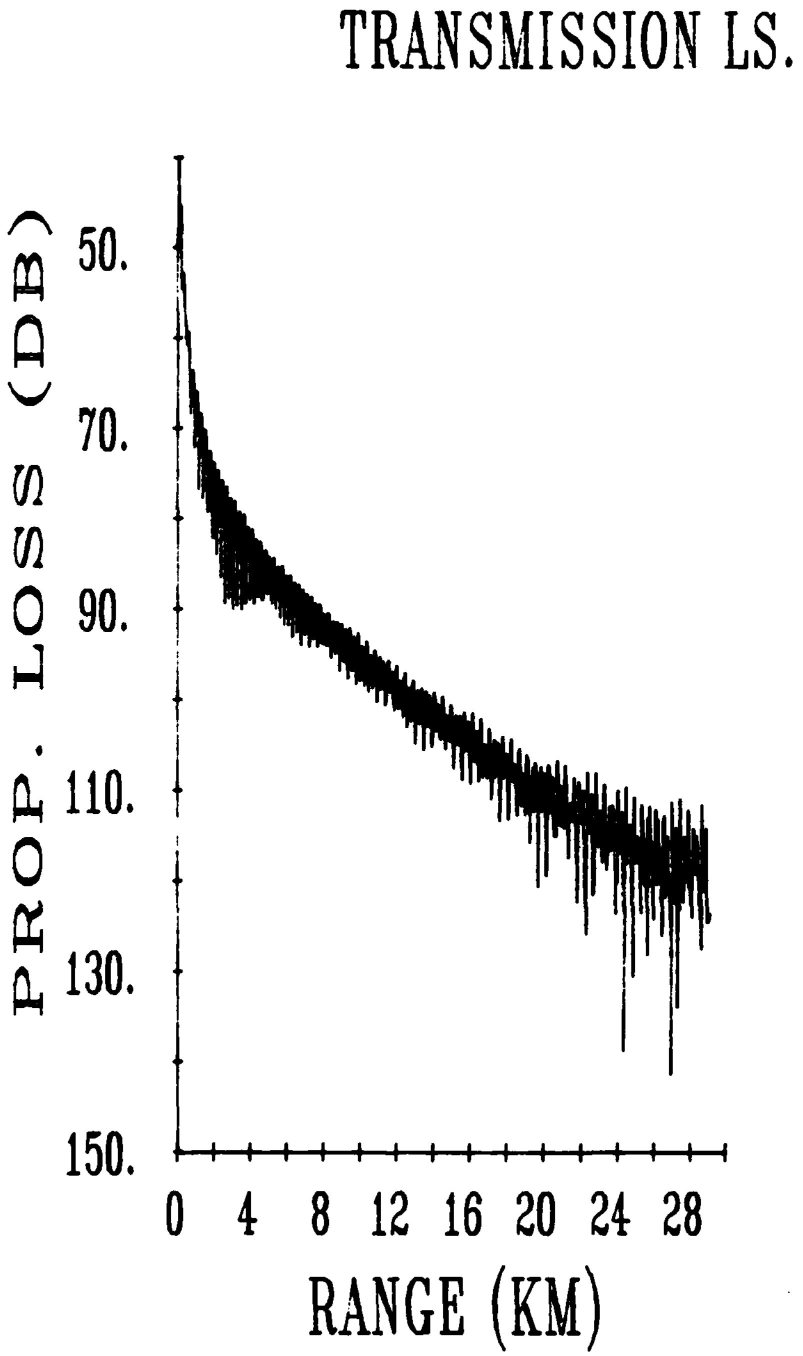
1 25.0 HZ L51H SM 0



2 25.0 HZ L51H SM 0

TEST CASE DATA

```
06  1                                     DATA FOR TLOSS51HH
 0.00      1500.0              .0000000             1.00
50.00      1500.0              .0000000             1.00
100.0      1500.0              .0000000             1.00
100.0      2800.0    1200.0    .00006286   .0001281 2.50
125.0      3000.0    1400.0    .00006286   .0001281 2.50
125.0      3000.0    1400.0    .00006286   .0001281 2.50
   2    3    02 1024   00  000    1   020.0    10000.    1000.0     8000.0
 0.0        1500.0   .0000000   1.00     0000.0        0    0   00     0
 00.       00.0      00.000       1
```



1 20.0 HZ L51H SM 0

IMPULS11B USER'S MANUAL

A. Introduction

IMPULS11B, our streamlined FFP impulse program, was completed in October of 1982. It is an extension of IMPULS10BB, the former interim standard impulse program.

IMPULS11B has these features and options for the user:

1. The FFP method of computation

2. A comprehensive selection of modelling parameters. These include:
   a. A detailed sound velocity profile
   b. Choice of source depth
   c. Choice of detector depth
   d. Choice of frequency band and source frequency
   e. Choice of wave number integrand limits
   f. Optional beaming, as it is affected by integrand tapering
   g. Optional surface roughness
   h. Choice of either fluid or vacuum upper half space
   i. Possibility of the source and detector in the fluid upper half space
   j. Optional internal roughness
   k. Choice of initial range, range spacing, and range count

3. Output. This includes:
   a. A printout of all input
   b. The pulse signal plots

4. Additional Optional Tables and Plots. These include:

   a. The wave integrand plots of the FFP for certain frequencies

5. Numerical output of the program for further processing or plotting. Formerly, this was called punched output, since it went out to the card punch.

B. Input Variables and Formats

Program input for IMPULS11B includes both the physical model and program control. This section of the manual will list and explain each line of the input.

1. Entry 1, one line

```
MMAXF, NOUTPT, NREVRS, NOISE, NCARD, NPLOT, NREAD      variable names
( 7I3 )                                                format
```

MMAXF is a control variable which is used to read in the correct number of points in the sound velocity profile. (see Entry 2. - next Entry)

NOUTPT is the number of output signal plots.

NREVRS is both a flag and a pointer. If non-zero, it signals that there will be pulse reversal, and it points to the output signal which is reversed and used as a source function.

NOISE is a control flag that, when set to zero, (be careful!) zeroes out the output signal which is outside the UMIN to UMAX interval. This eliminates the noise outside the specified time interval, which can be severe at long ranges.

NCARD is a control variable that, when set non-zero, allows the program to put out signal data on unit 7 in a form which is readable as input by another program. A zero value will prevent any output on unit 7. This is normally used to make graphic plots. (Originally, on the IBM 360/91, unit 7 was the punched card unit, hence the name NCARD.)

NPLOT is a control variable that, when set non-zero, allows the program to put out printer plots of the output signals.

NREAD is a control variable that, when set non-zero, directs the program to interpret Entry 2 data (see below) as thickness input instead of the usual depth input.

2. Entry 2, MMAXF lines (max. of 49 lines)

```
(( V(M,N), N=1,7 ), M=1,MMAXF )                        variable name
( 7D10.0 )                                             format
```

V is an array that contains the sound velocity profile. For each point in the profile, and on each line, (one point per line), the program expects:

a. a depth value, in meters, except when NREAD is set non-zero, in which case the program expects a thickness value, in meters
b. a sound (pressure) velocity value, in meters per second
c. a sound (shear) velocity value, in meters per second, if applicable
d. a corresponding attenuation value for pressure
e. a corresponding attenuation value for shear, if applicable
f. a density value, in grams per c.c., or specific density
g. a roughness value, if applicable

Note: No depth value can be less than a previous depth value.

3. Entry 3, one line

```
KSF, LRF, IRES, LS, KKK, IBEAM, NW, RSPACE, PO, C, CO
                                                       variable names
( 7I5, 4D10.0 )                                        format
```

KSF is a pointer to the source depth value. It points to one of the points in the sound velocity profile; the one which is the source depth value.

LRF is a pointer to the detector (or receiver) depth value. It's function is similar to KSF.

IRES is a control variable that, when set non-zero, requires the program to taper the response function at each end.

LS is the number of points used in the wave integration.

KKK is a control variable that indicates:
a. If 0 or 2, this is the last case.
b. If -1 or -2, the next case starts with Entry 1, (new profile).
c. If +1 or +3, the next case starts with Entry 3.
The second value in each case (2, -2, +3), will direct the program to take each signal and compress it back to its source. A plot of this compressed pulse follows each output signal.

IBEAM is a control variable that, when set non-zero, requires the program to taper the wave integrand at each end.

NW is a flag that, when set non-zero, directs the program to print out selected integrand plots.

RSPACE is the range spacing between output signals, in kilometers. However, the program can only approximate this value.

PO is the source level. For this program we normally use a value of 10000.

C is the lower bound sound velocity in the wave integration, in meters per second.

CO is the upper bound sound velocity in the wave integration, in meters per second.

4. Entry 4, one line
FREGO, NX, INTF, FREGM, DT, RO, FREGS, UMAX, UMIN   variable name
(F10.0, 2I5, 6F10.0)   format

FREGO is the lower bound of the frequency integration, in Hertz.

NX is the number of FFT points and the number of points in each output signal.

INTF is a variable that specifies how often, within the frequency integration, the wave integrand will be plotted on the printer, (every INTF'th frequency). Enter NX+1 for no wave integrand plots.

FREGM is the upper bound of the frequency integration, in Hertz.

DT is the time difference between the successive points in the output signals, in secs.

RO is the initial range or the range of the first output signal, in kilometers.

FREGS is the sound source frequency, in Hertz.

UMAX is a sound velocity (meters/sec.) that is used to specify the lower bound of the output signal if some of the signal is considered noise. (See NOISE, Entry 1, also see CRED, Entry 5

UMIN is similar in function to UMAX, but used to specify the upper bound.

5. Entry 5, one line
REFLTO, ALPHAO, ATTENO, RHOO, H, CRED variable names
(6D10.0) format

REFLTO is the RMS surface roughness value, in meters. If this value is non-zero, the program explicitly sets the next three values to zero.

ALPHAO is, if present, the sound velocity in the fluid upper half space, in meters per second. This is the variable that determines whether the present case has either a fluid or vacuum upper half space.

ATTENO is, if present, the attenuation in the fluid upper half space.

RHOO is, if present, the density of the fluid upper half space, in grams per c.c.

H is, if present, the distance between the source and detector in the fluid upper half space. To use this option:

a. The first two sound velocity depths must be zero.
b. There must be some variation in the pressure sound velocity profile.
c. There must not be any shear sound velocity.
d. ALPHAO and H must be present.

CRED is the reducing velocity. For each output signal, the starting time is offset from TO at range O (point of origin) by a time equal to that range divided by CRED. This value defaults to CO, for a blank or zero entry.

C. Operating Procedures

The primary computer at Lamont-Doherty Geological Observatory is a VAX 11/780, which runs under the VMS operating system. Listed below are four distinctly different operations that are used to process IMPULS11B.

1. Compiling-Linking

   a. Compiling (FORT command)
   Compiling the IMPULS11B program on the VAX 11/780 has always been done with the following switches on:

      1. /CHECK Forbids array pointer overflow.
      2. /NOI4 Integer variables are set to 2 bytes.
      3. /NOF77 Where the compiler needs to choose between Fortran 77 and Fortran 4+, the latter is chosen.
      4. /LIST A compiler listing is made and stored on the disk as IMPULS11B.LIS.

   b. Linking (LINK command)
   Linking is straight-forward. No extra routines are linked with IMPULS11B.

2. Running

   IMPULS11B is normally run through a batch stream, in particular the SLOW batch stream, since this is the cheapest route, especially at night. However, there is another batch stream with higher priority (and higher cost) that can be used for quicker results, the SYS$BATCH batch stream. In order to use either of these two batch streams:

   a. The executable file (IMPULS11B.EXE) must be ready to use. The LINK operation satisfies this requirement.

   b. The data file must also be ready to use. This is normally prepared by the user using the editor, (and the IMPULS11B USER'S MANUAL, if necessary) at a terminal.

   c. Submit the command file, IMPULS11B.COM, to the batch stream (or batch queue) using a command such as one of these:

```
SUBMIT/NAME=IEB15OCT/QUEUE=SLOW/AFTER=20:00 IMPULS11B
      SLOW queue, Job starts at 8:00 P.M.

SUBMIT/NAME=IEB15OCT/QUEUE=SLOW IMPULS11B
      SLOW queue, Job starts now

SUBMIT/NAME=IEB15OCT IMPULS11B
      SYS$BATCH queue ( the 'regular' queue ),
      Job starts now
```

   d. The following command file, also prepared using the editor, is a typical file submitted, using one of the previous submit commands.

```
$ASSIGN IEB15OCT SYS$PRINT                   (1)
$ASSIGN IEB15OCT SYS$OUTPUT                  (2)
$ASSIGN DRO:IEB15OCT.PLT FOR007              (3)
$ASSIGN DRO:IEB15OCT.OUT FOR006              (4)
$ASSIGN DRO:STANDARD.DAZ FOR005              (5)
$TYPE IMPULS11B                              (6)
$COPY DRO:STANDARD.DAZ DRO:IEB15OCT.DAT      (7)
$RUN IMPULS11B                               (8)
```

This run is called the IEB15OCT run. 'IEB' stands for IMPULSE ELEVEN B. '15OCT' indicates that this is the 15th run in October.

The first two lines in this command file put the log file on the disk instead of the printer.
The third line puts unit 7 output into file IEB15OCT.PLT.
The fourth line puts unit 6 output into file IEB15OCT.OUT.
The fifth line draws unit 5 input from STANDARD.DAZ.
The sixth line copies IMPULS11B.LIS to IEB15OCT.LIS.
The seventh line copies STANDARD.DAZ to IEB15OCT.DAT.
The eighth line runs program IMPULS11B.

A command file such as this yields a file group; each file having the same name but a different extension. These files are then treated collectively. They are printed together and stored on tape together. Such a group is, for example:

IEB15OCT.DAT	a copy of the data file
IEB15OCT.LIS	the program listing file
IEB15OCT.LOG	the log file
IEB15OCT.OUT	the output file
IEB15OCT.PLT	the numerical output file

The user normally gets a complete printout of the run by typing the command:

```
PRINT IEB15OCT.*;*
```

on the terminal.

3. Copying Runs to Tape (Archiving)

Periodically, it is necessary to clear the old runs off the disk and put them on tape. This is a very simple operation and proceeds as follows:

a. Initialize a tape. Give it a name that is somehow indicative of the present month. For example:

```
INITIALIZE/DENSITY=1600 MT1: OCTNOV
```

This tape would have runs made in October and November. This step is not necessary if the user is adding runs to a tape that is already partly filled. In such a case, be careful not to initialize!

b. Mount the tape with the MOUNT command.

```
MOUNT MT1: OCTNOV
```

This is not the same as physically mounting the tape on the tape drive, which is assumed to have taken place previously.

c. Copy all the run files from disk to tape with the COPY command. For example in this case:

```
COPY *OCT.*;* MT1:
COPY *NOV.*;* MT1:
```

would copy all the October and all the November runs onto the tape mounted on tape drive 1.

d. For future reference, it is good to get a list of names of the files on the tape. This is accomplished as follows:

```
DIRECTORY/PRINT MT1:
```

The printout resulting from this command does not have the tape name on it, so the user should write this on immediately.
If, for some reason, this printout is lost, the tape can be remounted and the directory can be taken again. But this is no reason to be careless with the original directory printout.

e. Dismount the tape with the dismount command:

```
DISMOUNT MT1:
```

f. Delete the files on the disk that have just been copied to the tape. In this case:

```
DELETE *OCT.*;*
DELETE *NOV.*;*
```

4. Distributing the Program by Tape

The program and test cases are distributed using tapes with the same format as the archival tapes. This is the Files-11 format, peculiar to Digital Equipment Corporation. The tapes from this system have a density of 1600 bpi, and a blocksize of 2048 bytes.

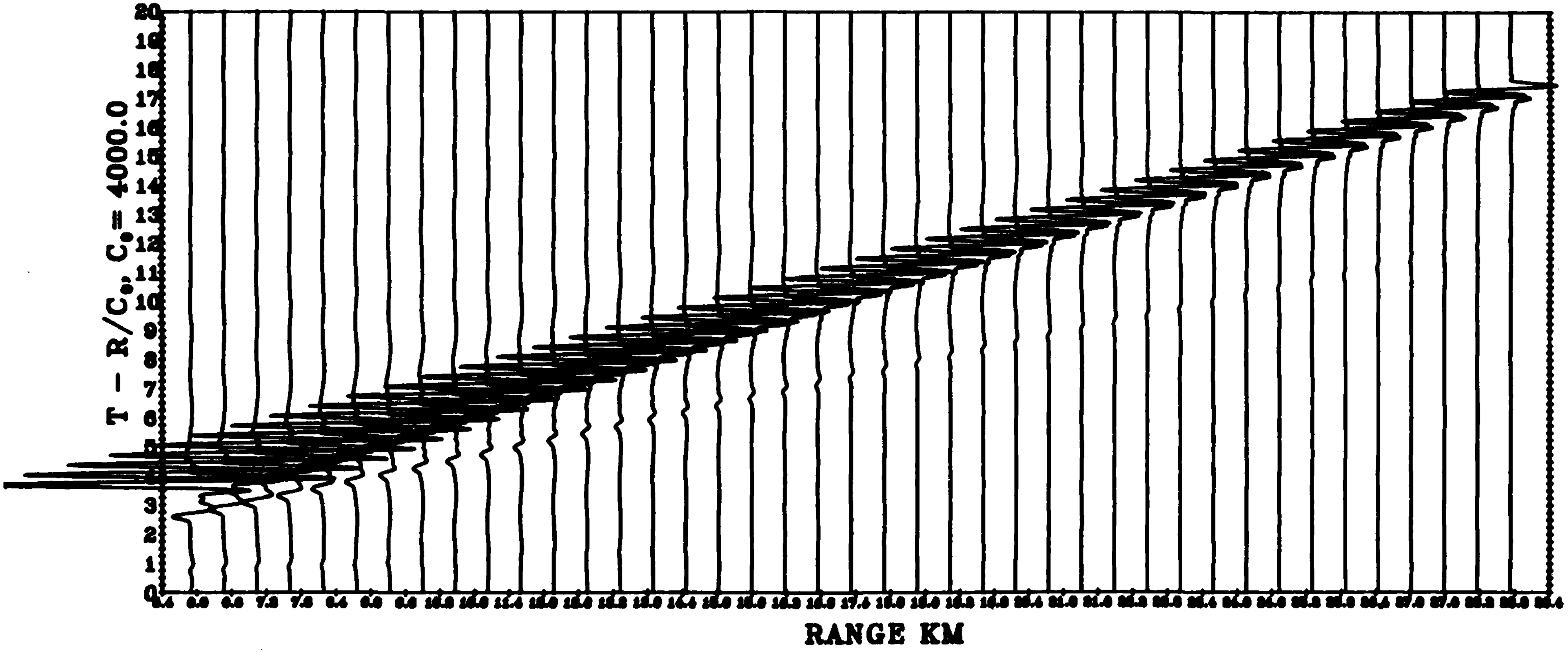
If any further questions arise, feel free to call:

Dr. H. Kutschale	(914) 359-2900 x382
Mr. L. Carroll	(914) 359-2900 x221

TEST CASE DATA

```
06 41  0  1  1  1  0  0              DATA FOR IMPULS11G
 0.00     1500.0             .00000150             1.00
50.00     1500.0             .00000150             1.00
100.0     1500.0             .00000150             1.00
100.0     3000.0    1400.0   .00000770  .000016    2.50
125.0     3000.0    1400.0   .00000770  .000016    2.50
125.0     3000.0    1400.0   .00000770  .000016    2.50
   2    3   00 2048   00  001    1   00.50   10000.    1000.0    8000.0
0.5000000 2048   60 4.1000000 0.010    05.90     2.0       7500.0    1500.0
0.0       1500.0   .0000150   1.00     0000.0    4000.0       00    0     1
```

PRESSURE


T - R/C_e, C_e= 4000.0
0
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
5.4 6.0 6.6 7.2 7.8 8.4 9.0 9.6 10.2 10.8 11.4 12.0 12.6 13.2 13.8 14.4 15.0 15.6 16.2 16.8 17.4 18.0 18.6 19.2 19.8 20.4 21.0 21.6 22.2 22.8 23.4 24.0 24.6 25.2 25.8 26.4 27.0 27.6 28.2 28.8 29.4
RANGE KM

IMPULS22B USER'S MANUAL

A. Introduction

IMPULS22B, one of our streamlined FFP impulse programs, was completed in March of 1983. It is an extension of IMPULS11B, one of the interim standard impulse programs.

IMPULS22B has these features and options for the user:

1. The FFP method of computation
   a. The complete, unabridged version
   b. An abbreviated version to calculate the effect of one particular number of bounces

2. A comprehensive selection of modelling parameters. These include:
   a. A detailed sound velocity profile
   b. Choice of source depth
   c. Choice of detector depth
   d. Choice of frequency band and source frequency
   e. Choice of wave number integrand limits
   f. Optional beaming, as it is affected by integrand tapering
   g. Optional surface roughness
   h. Choice of either fluid or vacuum upper half space
   i. Possibility of the source and detector in the fluid upper half space
   j. Optional internal roughness
   k. Choice of initial range, range spacing, and range count

3. Output. This includes:
   a. A printout of all input
   b. The pulse signal plots

4. Additional Optional Tables and Plots. These include:

   a. The wave integrand plots of the FFP for certain frequencies

5. Numerical output of the program for further processing or plotting. Formerly, this was called punched output, since it went out to the card punch.

B. Input Variables and Formats

Program input for IMPULS22B includes both the physical model and program control. This section of the manual will list and explain each line of the input.

1. Entry 1, one line

```
  MMAXF, NOUTPT, NREVRS, NOISE, NCARD, NPLOT, NREAD     variable names
   ( 7I3 )                                              format
```

MMAXF is a control variable which is used to read in the correct number of points in the sound velocity profile. (see Entry 2. - next Entry)

NOUTPT is the number of output signal plots.

NREVRS is both a flag and a pointer. If non-zero, it signals that there will be pulse reversal, and it points to the output signal which is reversed and used as a source function.

NOISE is a control flag that, when set to zero, (be careful!) zeroes out the output signal which is outside the UMIN to UMAX interval. This eliminates the noise outside the specified time interval, which can be severe at long ranges.

NCARD is a control variable that, when set non-zero, allows the program to put out signal data on unit 7 in a form which is readable as input by another program. A zero value will prevent any output on unit 7. This is normally used to make graphic plots. (Originally, on the IBM 360/91, unit 7 was the punched card unit, hence the name NCARD.)

NPLOT is a control variable that, when set non-zero, allows the program to put out printer plots of the output signals.

NREAD is a control variable that, when set non-zero, directs the program to interpret Entry 2 data (see below) as thickness input instead of the usual depth input.

2. Entry 2, MMAXF lines (max. of 49 lines)

```
  (( V(M,N), N=1,7 ), M=1,MMAXF )                      variable name
  ( 7D10.0 )                                           format
```

V is an array that contains the sound velocity profile. For each point in the profile, and on each line, (one point per line), the program expects:

a. a depth value, in meters, except when NREAD is set non-zero, in which case the program expects a thickness value, in meters

b. a sound (pressure) velocity value, in meters per second

c. a sound (shear) velocity value, in meters per second, if applicable

d. a corresponding attenuation value for pressure

e. a corresponding attenuation value for shear, if applicable

f. a density value, in grams per c.c., or specific density

g. a roughness value, if applicable

Note: No depth value can be less than a previous depth value.

3. Entry 3, one line

```
  KSF, LRF, IRES, LS, KKK, IBEAM, NW, RSPACE, PO, C, CO
                                                       variable names
  ( 7I5, 4D10.0 )                                      format
```

KSF is a pointer to the source depth value. It points to one of the points in the sound velocity profile; the one which is the source depth value.

LRF is a pointer to the detector (or receiver) depth value. It's function is similar to KSF.

IRES is a control variable that, when set non-zero, requires the program to taper the response function at each end.

LS is the number of points used in the wave integration.

KKK is a control variable that indicates:
a. If 0 or 2, this is the last case.
b. If -1 or -2, the next case starts with Entry 1, (new profile).
c. If +1 or +3, the next case starts with Entry 3.
The second value in each case (2, -2, +3), will direct the program to take each signal and compress it back to its source. A plot of this compressed pulse follows each output signal.

IBEAM is a control variable that, when set non-zero, requires the program to taper the wave integrand at each end.

NW is a flag that, when set non-zero, directs the program to print out selected integrand plots.

RSPACE is the range spacing between output signals, in kilometers. However, the program can only approximate this value.

PO is the source level. For this program we normally use a value of 10000.

C is the lower bound sound velocity in the wave integration, in meters per second.

CO is the upper bound sound velocity in the wave integration, in meters per second.

4. Entry 4, one line
FREQO, NX, INTF, FREQM, DT, RO, FREQS, UMAX, UMIN   variable names
(F10.0, 2I5, 6F10.0)   format

FREQO is the lower bound of the frequency integration, in Hertz.

NX is the number of FFT points and the number of points in each output signal.

INTF is a variable that specifies how often, within the frequency integration, the wave integrand will be plotted on the printer, (every INTF'th frequency). Enter NX+1 for no wave integrand plots.

FREQM is the upper bound of the frequency integration, in Hertz.

DT is the time difference between the successive points in the output signals, in secs.

RO is the initial range or the range of the first output signal, in kilometers.

FREQS is the sound source frequency, in Hertz.

UMAX is a sound velocity (meters/sec.) that is used to specify the lower bound of the output signal if some of the signal is considered noise. (See NOISE, Entry 1, also see CRED, Entry 5)

UMIN is similar in function to UMAX, but used to specify the upper bound.

5. Entry 5, one line
REFLTO, ALPHAO, ATTENO, RHOO, H, CRED, BOUNCE variable name
(6D10.0, I5) format

REFLTO is the RMS surface roughness value, in meters. If this value is non-zero, the program explicitly sets the next three values to zero.

ALPHAO is, if present, the sound velocity in the fluid upper half space, in meters per second. This is the variable that determines whether the present case has either a fluid or vacuum upper half space.

ATTENO is, if present, the attenuation in the fluid upper half space.

RHOO is, if present, the density of the fluid upper half space, in grams per c.c.

H is, if present, the distance between the source and detector in the fluid upper half space. To use this option:

a. The first two sound velocity depths must be zero.
b. There must be some variation in the pressure sound velocity profile.
c. There must not be any shear sound velocity.
d. ALPHAO and H must be present.

CRED is the reducing velocity. For each output signal, the starting time is offset from TO at range 0 (point of origin) by a time equal to that range divided by CRED. This value defaults to CO, for a blank or zero entry.

BOUNCE is a control variable that, when set non-zero, directs the program to calculate only the effects of a certain number of bounces, BOUNCE being that number. This is somewhat similar to the 'H variable' output and should not be used in conjunction with that feature.

C. Operating Procedures

The primary computer at Lamont-Doherty Geological Observatory is a VAX 11/780, which runs under the VMS operating system. Listed below are four distinctly different operations that are used to process IMPULS22B.

1. Compiling-Linking

   a. Compiling (FORT command)

   Compiling the IMPULS22B program on the VAX 11/780 has always been done with the following switches on:

   1. /CHECK Forbids array pointer overflow.
   2. /NOI4 Integer variables are set to 2 bytes.
   3. /NOF77 Where the compiler needs to choose between Fortran 77 and Fortran 4+, the latter is chosen.
   4. /LIST A compiler listing is made and stored on the disk as IMPULS22B.LIS.

   b. Linking (LINK command)

   Linking is straight-forward. No extra routines are linked with IMPULS22B.

2. Running

IMPULS22B is normally run through a batch stream, in particular the SLOW batch stream, since this is the cheapest route, especially at night. However, there is another batch stream with higher priority (and higher cost) that can be used for quicker results, the SYS$BATCH batch stream. In order to use either of these two batch streams:

a. The executable file (IMPULS22B.EXE) must be ready to use. The LINK operation satisfies this requirement.

b. The data file must also be ready to use. This is normally prepared by the user using the editor, (and the IMPULS22B USER'S MANUAL, if necessary) at a terminal.

c. Submit the command file, IMPULS22B.COM, to the batch stream (or batch queue) using a command such as one of these:

```
SUBMIT/NAME=ITB17MAR/QUEUE=SLOW/AFTER=20:00 IMPULS22B
       SLOW queue, Job starts at 8:00 P.M.

SUBMIT/NAME=ITB17MAR/QUEUE=SLOW IMPULS22B
       SLOW queue, Job starts now

SUBMIT/NAME=ITB17MAR IMPULS22B
       SYS$BATCH queue ( the 'regular' queue ),
       Job starts now
```

d. The following command file, also prepared using the editor, is a typical file submitted, using one of the previous submit commands.

```
$ASSIGN ITB17MAR SYS$PRINT                  (1)
$ASSIGN ITB17MAR SYS$OUTPUT                 (2)
$ASSIGN DRO:ITB17MAR.PLT FOR007             (3)
$ASSIGN DRO:ITB17MAR.OUT FOR006             (4)
$ASSIGN DRO:STRATE.DAZ FOR005               (5)
$TYPE IMPULS22B                             (6)
$COPY DRO:STRATE.DAZ DRO:ITB17MAR.DAT       (7)
$RUN IMPULS22B                              (8)
```

This run is called the ITB17MAR run. 'ITB' stands for IMPULSE TWENTY TWO B. '17MAR' indicates that this is the 17th run in March.

The first two lines in this command file put the log file on the disk instead of the printer.
The third line puts unit 7 output into file ITB17MAR.PLT.
The fourth line puts unit 6 output into file ITB17MAR.OUT.
The fifth line draws unit 5 input from STRATE.DAZ.
The sixth line copies IMPULS22B.LIS to ITB17MAR.LIS.
The seventh line copies STRATE.DAZ to ITB17MAR.DAT.
The eighth line runs program IMPULS22B.

A command file such as this yields a file group; each file having the same name but a different extension. These files are then treated collectively. They are printed together and stored on tape together. Such a group is, for example:

ITB17MAR.DAT	a copy of the data file
ITB17MAR.LIS	the program listing file
ITB17MAR.LOG	the log file
ITB17MAR.OUT	the output file
ITB17MAR.PLT	the numerical output file

The user normally gets a complete printout of the run by typing the command:

```
PRINT ITB17MAR.*;*
```

on the terminal.

3. Copying Runs to Tape (Archiving)

Periodically, it is necessary to clear the old runs off the disk and put them on tape. This is a very simple operation and proceeds as follows:

a. Initialize a tape. Give it a name that is somehow indicative of the present month. For example:

```
INITIALIZE/DENSITY=1600 MT1: FEBMAR
```

This tape would have runs made in February and March. This step is not necessary if the user is adding runs to a tape that is already partly filled. In such a case, be careful not to initialize!

b. Mount the tape with the MOUNT command.

```
MOUNT MT1: FEBMAR
```

This is not the same as physically mounting the tape on the tape drive, which is assumed to have taken place previously.

c. Copy all the run files from disk to tape with the COPY command. For example in this case:

```
COPY *FEB.*;* MT1:
COPY *MAR.*;* MT1:
```

would copy all the February and all the March runs onto the tape mounted on tape drive 1.

d. For future reference, it is good to get a list of names of the files on the tape. This is accomplished as follows:

```
DIRECTORY/PRINT MT1:
```

The printout resulting from this command does not have the tape name on it, so the user should write this on immediately.
If, for some reason, this printout is lost, the tape can be remounted and the directory can be taken again. But this is no reason to be careless with the original directory printout.

e. Dismount the tape with the dismount command:

```
DISMOUNT MT1:
```

f. Delete the files on the disk that have just been copied to the tape. In this case:

```
DELETE *FEB.*;*
DELETE *MAR.*;*
```

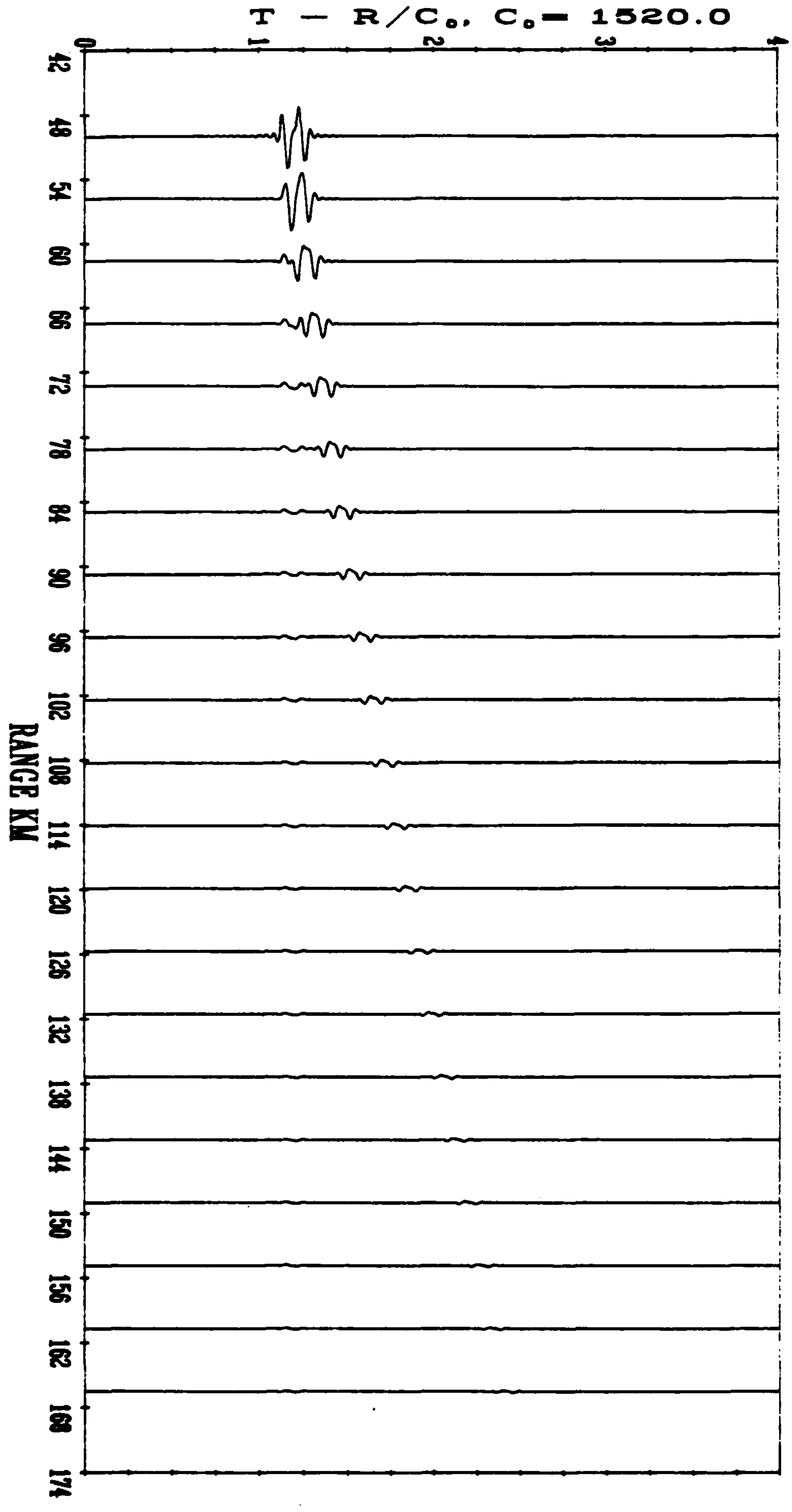
4. Distributing the Program by Tape

The program and test cases are distributed using tapes with the same format as the archival tapes. This is the Files-11 format, peculiar to Digital Equipment Corporation. The tapes from this system have a density of 1600 bpi, and a blocksize of 2048 bytes.

If any further questions arise, feel free to call:

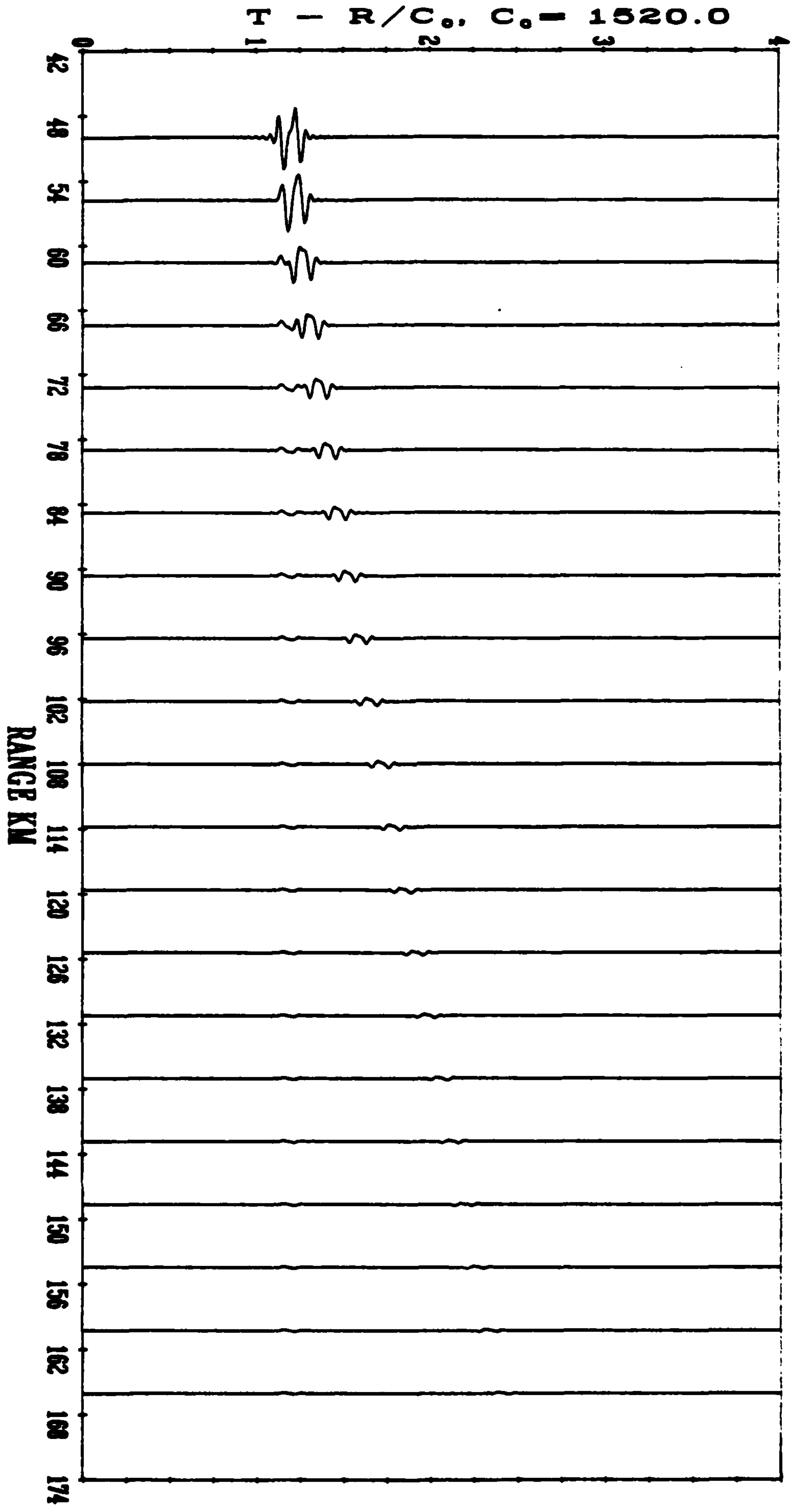
Dr. H. Kutschale	(914) 359-2900 x382
Mr. L. Carroll	(914) 359-2900 x221

```
ITB15MAR.DAT;3                          6-APR-1983 15:29
                  TEST CASE DATA

04 21  0  1  1  1  0
000.0     1490.0             .000005             1.03
050.0     1490.0             .000005             1.03
4200.     1490.0             .000005             1.03
4200.     1520.0             .000010             1.03
   2   02    0  256    0  00  001   05.    10000.0    1488.0    1519.999
2.441407   512   40 24.414063 0.0080   50.0      8.0
0.0        0.00     00.0      00.00     000.0     1519.999    1    0    1
```

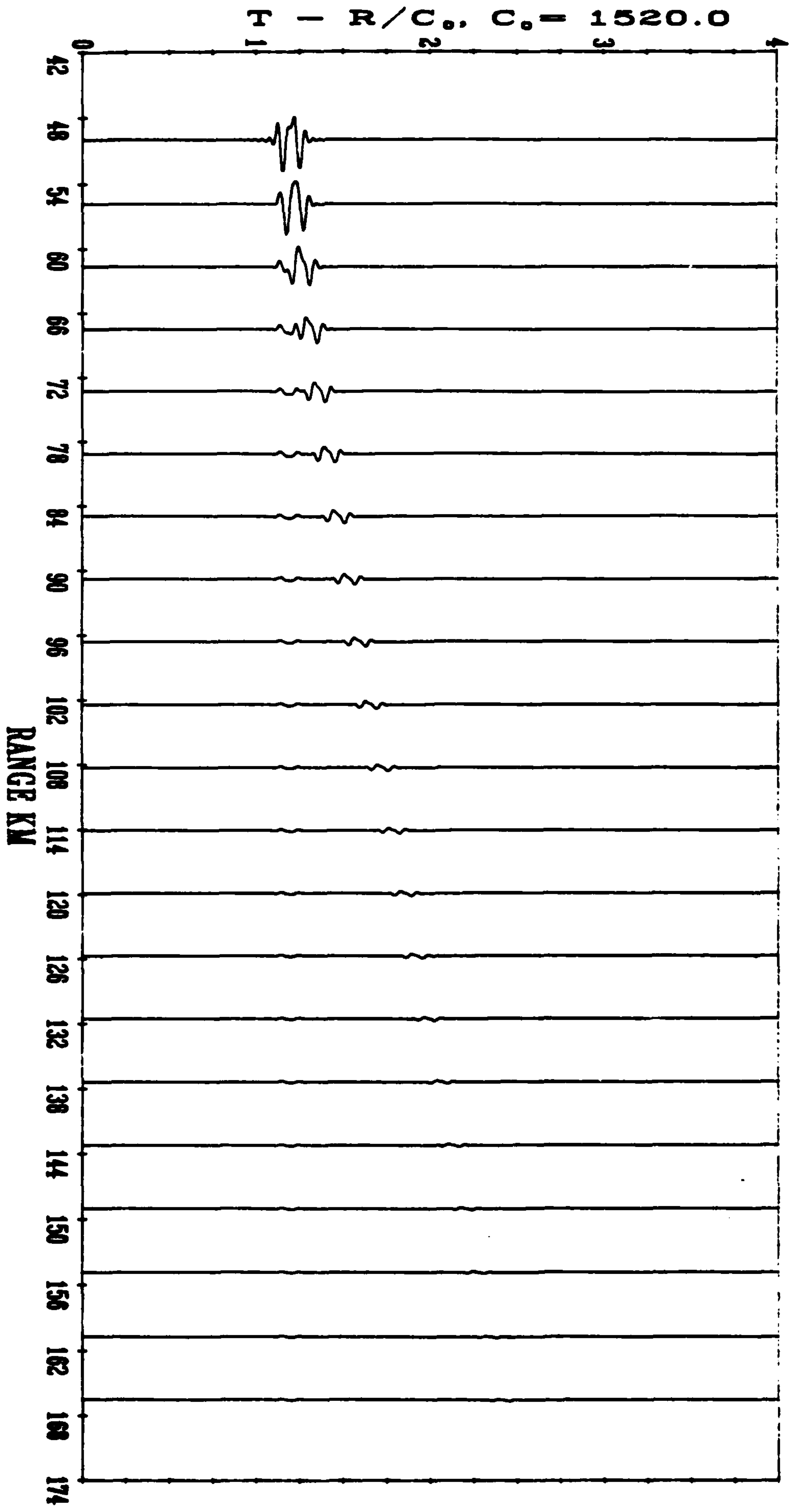

T — R/C₀, C₀= 1520.0
0
1
2
3
4
42
48
54
60
66
72
78
84
90
96
102
108
114
120
126
132
138
144
150
156
162
168
174
RANGE KM

TEST CASE DATA

```
04 21  0  1  1  1  0
000.0     1490.0              .000005             1.03
050.0     1490.0              .000005             1.03
4200.     1490.0              .000005             1.03
4200.     1520.0              .000010             1.03
   2   02     0  128    0   00 001   05.    10000.0    1488.0    1519.999
2.441407   512   40 24.414063 0.0080    50.0      8.0
0.0        0.00      00.0      00.00     000.0     1519.999    1    0    1
```

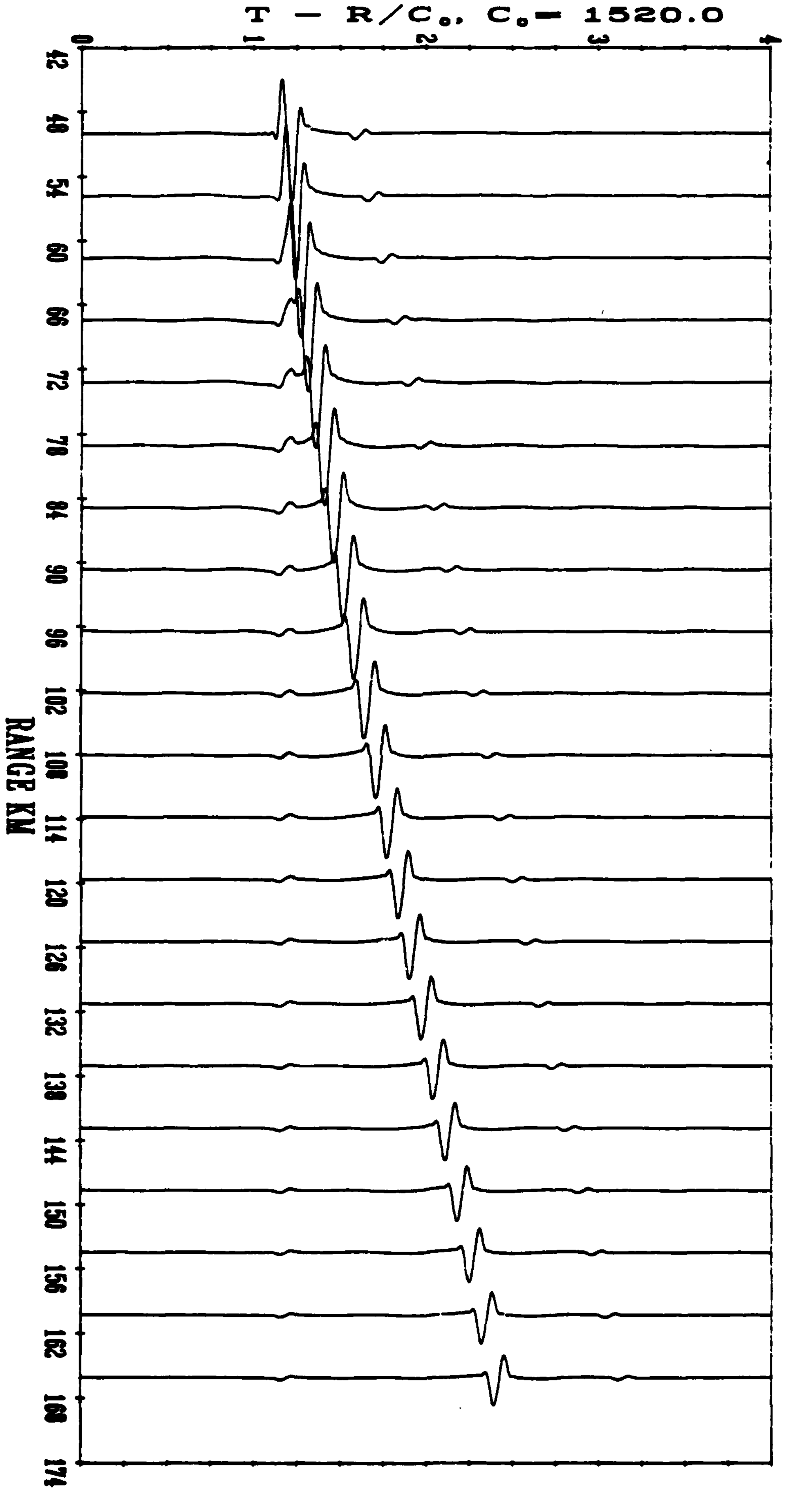

T − R/C₀, C₀= 1520.0
0
1
2
3
4
42
48
54
60
66
72
78
84
90
96
102
108
114
120
126
132
138
144
150
156
162
168
174
RANGE KM

```
ITB18MAR.DAT;2                              6-APR-1983 15:29

                  TEST CASE DATA

04 21  0  1  1  1  0
000.0     1490.0              .000005             1.03
050.0     1490.0              .000005             1.03
4200.     1490.0              .000005             1.03
4200.     1520.0     600.0    .000010             1.03
   2   02     0  256    0   00  001   05.    10000.0    1488.0    1519.999
2.441407  512    40 24.414063 0.0080    50.0     8.0
0.0        0.00      00.0      00.00     000.0     1519.999     1   0    1
```


T − R/C₀, C₀= 1520.0
0
1
2
3
4
42
48
54
60
66
72
78
84
90
96
102
108
114
120
126
132
138
144
150
156
162
168
174
RANGE KM

TEST CASE DATA

```
04 21   0   1   1   1   0
000.0       1490.0                  .000005             1.03
000.0       1490.0                  .000005             1.03
4200.       1490.0                  .000005             1.03
4200.       1520.0                  .000010             1.03
   2    02     0   128     0     00  001     05.     10000.0
2.441407    512    40  24.414063 0.0080      50.0    8.0     1488.0      1519.999
0.0         1490.0     .000005   01.03       25.00          1519.999     0     0     1
```


T — R/C₀, C₀= 1520.0
0
1
2
3
4
42
48
54
60
66
72
78
84
90
96
102
108
114
120
126
132
138
144
150
156
162
168
174
RANGE KM

IMPULS11G USER'S MANUAL

A. Introduction

IMPULS11G, our more complete impulse program, was completed in October of 1982. It is an extension of IMPULS10G, the former interim standard impulse program.

IMPULS11G has these features and options for the user:

1. A choice of computational methods;
   - a. The FFP method
   - b. The Normal Mode method

2. Optional horizontal or vertical displacement and optional horizontal or vertical particle velocity in addition to the usual pressure values for output pulse signals.

3. A comprehensive selection of modelling parameters. These include:
   - a. A detailed sound velocity profile
   - b. Choice of source depth
   - c. Choice of detector depth
   - d. Choice of frequency band and source frequency
   - e. Choice of wave number integrand limits
   - f. Optional beaming, as it is affected by integrand tapering (in the FFP mode)
   - g. Optional surface roughness
   - h. Choice of either fluid or vacuum upper half space
   - i. Possibility of the source and detector in the fluid upper half space
   - j. Optional internal roughness
   - k. Choice of initial range, range spacing, and range count

4. Output. This includes:
   - a. A printout of all input
   - b. The pulse signal plots (usually)
   - c. A transmission loss plot for the range of the pulse signals (Normal Mode method only)

5. Additional Optional Tables and Plots. These include:
   - a. Dispersion curves, which includes: Phase velocities and group velocities
   - b. The wave integrand plots of the FFP for certain frequencies

   Note: All short plots except the wave integrands are repeated, the repeated plot being a smoothed copy of the first and having the same bounds as the first.

6. Numerical output of the program for further processing or plottin Formerly, this was called punched output, since it went out to the card punch.

B. Input Variables and Formats

Program input for IMPULS11G includes both the physical model and program control. This section of the manual will list and explain each line of the input.

1. Entry 1, one line

```
MMAXF, NOUTPT, NREVRS, NOISE, NCARD, NPLOT          variable names
( 6I3 )                                              format
```

MMAXF is a control variable which is used to read in the correct number of points in the sound velocity profile. (see Entry 2. - next Entry)

NOUTPT is the number of output signal plots

NREVRS is both a flag and a pointer. If non-zero, it signals that there will be pulse reversal, and it points to the output signal which is reversed and used as a source function.

NOISE is a control flag that, when set to zero, (be careful!) zeroes out the output signal which is outside the UMIN to UMAX interval. This eliminates the noise outside the specified time interval, which can be severe at long ranges for the FFP mode.

NCARD is a control variable that, when set non-zero, allows the program to put out signal data on unit 7 in a form which is readable as input by another program. A zero value will prevent any output on unit 7. This is normally used to make graphic plots. (Originally, on the IBM 360/91, unit 7 was the punched card unit, hence the name NCARD.)

NPLOT is a control variable that, when set non-zero, allows the program to put out printer plots of the output signals.

2. Entry 2, MMAXF lines (max. of 49 lines)

```
(( V(M,N), N=1,7 ), M=1,MMAXF )                      variable name
( 7D10.0 )                                           format
```

V is an array that contains the sound velocity profile. For each point in the profile, and on each line, (one point per line), the program expects:
a. a depth value, in meters
b. a sound (pressure) velocity value, in meters per second
c. a sound (shear) velocity value, in meters per second, if applicable
d. a corresponding attenuation value for pressure
e. a corresponding attenuation value for shear, if applicable
f. a density value, in grams per c.c., or specific density
g. a roughness value, if applicable

Note: No depth value can be less than a previous depth value. The program will stop if this is so.

3. Entry 3, one line

```
KSF, LRF, IRES, LS, KKK, IBEAM, NM, RSPACE, PO, C, CO
                                       variable names
( 7I5, 4D10.0 )                        format
```

KSF is a pointer to the source depth value. It points to one of the points in the sound velocity profile; the one which is the source depth value.

LRF is a pointer to the detector (or receiver) depth value. It's function is similar to KSF.

IRES is a control variable that, when set non-zero, requires the program to taper the response function at each end.

LS is the number of points used in the wave integration.

KKK is a control variable that indicates:
a. If 0 or 2, this is the last case.
b. If -1 or -2, the next case starts with Entry 1, (new profile).
c. If +1 or +3, the next case starts with Entry 3.
The second value in each case (2, -2, +3), will direct the program to take each signal and compress it back to its source. A plot of this compressed pulse follows each output signal.

IBEAM is a control variable that, when set non-zero, requires the program to taper the wave integrand at each end.

NM is the maximum number of modes needed for the Normal Mode case.

RSPACE is the range spacing between output signals, in kilometers. For the Normal Mode method, this number is exact. For the FFP method, the program approximates this value.

PO is the source level. For this program we normally use a value of 10000.0

C is the lower bound sound velocity in the wave integration, in meters per second.

CO is the upper bound sound velocity in the wave integration, in meters per second.

4. Entry 4, one line

```
FREQO, NX, INTF, FREQM, DT, RO, FREQS, UMAX, UMIN
                                        variable names
( F10.0, 2I5, 6F10.0 )                  format
```

FREQO is the lower bound of the frequency integration, in Hertz.

NX is the number of FFT points and the number of points in each output signal.

INTF is a variable that specifies how often, within the frequency integration, the wave integrand will be plotted on the printer, (every INTF'th frequency). Enter NX+1 for no wave integrand plots.

FREQM is the upper bound of the frequency integration, in Hertz.

DT is the time difference between the successive points in the output signals, in secs.

RO is the initial range or the range of the first output signal, in kilometers.

FREQS is the sound source frequency, in Hertz.

UMAX is a sound velocity (meters/sec.) that is used to specify the lower bound of the output signal if some of the signal is considered noise. (See NOISE, Entry 1, also see CRED, Entry 5 .

UMIN is similar in function to UMAX, but used to specify the upper bound.

5. Entry 5, one line

```
REFLTO, ALPHAO, ATTENO, RHOO, H, CRED, KHPV, IDISP, KURVE,
IVEL                                                variable names
( 6D10.0, 4I5 )                                     format
```

REFLTO is the RMS surface roughness value, in meters. If this value is non-zero, the program explicitly sets the next three values to zero.

ALPHAO is, if present, the sound velocity in the fluid upper half space, in meters per second. This is the variable that determines whether the present case has either a fluid or vacuum upper half space.

ATTENO is, if present, the attenuation in the fluid upper half space.

RHOO is, if present, the density of the fluid upper half space, in grams per c.c.

H is, if present, the distance between the source and detector in the fluid upper half space. To use this option:

a. The first two sound velocity depths must be zero.
b. There must be some variation in the pressure sound velocity profile.
c. There must not be any shear sound velocity.
d. ALPHAO and H must be present.

CRED is the reducing velocity. For each output signal, the starting time is offset from TO at range 0 (point of origin) by a time equal to that range divided by CRED. This value defaults to CO, for a blank or zero entry.

KHPV is the control variable that determines whether output will be:

a. pressure, if zero.
b. vertical displacement, if positive.
c. horizontal displacement, if negative.

IVEL modifies this if it is set. (see below)
FFP only.

IDISP is a control variable that directs the program to forego output signals in favor of dispersion curves. In this case, the dispersion curve plot runs from FREGO to FREGM (min to max freq.) (entry 4) and from C to CO (min to max sound velocities) (entry 3). This option will also print phase velocity and group velocity plots for up to 5 modes, and punch out these phase and group values for further processing.

KURVE is a control variable that, when set non-zero, signals the program to contour the wave integrands. This produces a plot similar to the dispersion curve plot above while still allowing signal output. FFP only.

IVEL is a control variable that, when set non-zero, alters the output signal definition from displacement to velocity. KHPV must be set, or this has no effect. FFP only.

Note:

The program determines the computational method, (whether FFP or Normal Mode), by examining all of the attenuation values and the surface roughness value. If they are all zero, the Normal Mode method is selected.

C. Operating Procedures

The primary computer at Lamont-Doherty Geological Observatory is a VAX 11/780, which runs under the VMS operating system. Listed below are four distinctly different operations that are used to process IMPULS11G.

1. Compiling-Linking
   a. Compiling (FORT command)
   Compiling the IMPULS11G program on the VAX 11/780 has always been done with the following switches on:
      1. /CHECK Forbids array pointer overflow.
      2. /NOI4 Integer variables are set to 2 bytes.
      3. /NOF77 Where the compiler needs to choose between Fortran 77 and Fortran 4+, the latter is chosen.
      4. /LIST A compiler listing is made and stored on the disk as IMPULS11G.LIS.

   b. Linking (LINK command)
   Linking is straight-forward. No extra routines are linked with IMPULS11G.

2. Running
   IMPULS11G is normally run through a batch stream, in particular the SLOW batch stream, since this is the cheapest route, especially at night. However, there is another batch stream with higher priority (and higher cost) that can be used for quicker results, the SYS$BATCH batch stream. In order to use either of these two batch streams:

   a. The executable file (IMPULS11G.EXE) must be ready to use. The LINK operation satisfies this requirement.

   b. The data file must also be ready to use. This is normally prepared by the user using the editor, (and the IMPULS11G USER'S MANUAL, if necessary) at a terminal.

   c. Submit the command file, IMPULS11G.COM, to the batch stream (or batch queue) using a command such as one of these:

```
SUBMIT/NAME=IEG15OCT/QUEUE=SLOW/AFTER=20:00 IMPULS11G
      SLOW queue, Job starts at 8:00 P.M.

SUBMIT/NAME=IEG15OCT/QUEUE=SLOW IMPULS11G
      SLOW queue, Job starts now

SUBMIT/NAME=IEG15OCT IMPULS11G
      SYS$BATCH queue ( the 'regular' queue ),
      Job starts now
```

   d. The following command file, also prepared using the editor, is a typical file submitted, using one of the previous submit commands.

```
$ASSIGN IEG15OCT SYS$PRINT                   (1)
$ASSIGN IEG15OCT SYS$OUTPUT                  (2)
$ASSIGN DR0:IEG15OCT.PLT FOR007              (3)
$ASSIGN DR0:IEG15OCT.OUT FOR006              (4)
$ASSIGN DR0:STANDARD.DAZ FOR005              (5)
$TYPE IMPULS11G                              (6)
$COPY DR0:STANDARD.DAZ DR0:IEG15OCT.DAT      (7)
$RUN IMPULS11G                               (8)
```

This run is called the IEG15OCT run. 'IEG' stands for IMPULSE ELEVEN G. '15OCT' indicates that this is the 15th run in October.

The first two lines in this command file put the log file on the disk instead of the printer.
The third line puts unit 7 output into file IEG15OCT.PLT.
The fourth line puts unit 6 output into file IEG15OCT.OUT.
The fifth line draws unit 5 input from STANDARD.DAZ.
The sixth line copies IMPULS11G.LIS to IEG15OCT.LIS.
The seventh line copies STANDARD.DAZ to IEG15OCT.DAT.
The eighth line runs program IMPULS11G.

A command file such as this yields a file group; each file having the same name but a different extension. These files are then treated collectively. They are printed together and stored on tape together. Such a group is, for example:

IEG15OCT.DAT	a copy of the data file
IEG15OCT.LIS	the program listing file
IEG15OCT.LOG	the log file
IEG15OCT.OUT	the output file
IEG15OCT.PLT	the numerical output file

The user normally gets a complete printout of the run by typing the command:

```
PRINT IEG15OCT.*;*
```

on the terminal.

3. Copying Runs to Tape (Archiving)

Periodically, it is necessary to clear the old runs off the disk and put them on tape. This is a very simple operation and proceeds as follows:

a. Initialize a tape. Give it a name that is somehow indicative of the present month. For example:

```
INITIALIZE/DENSITY=1600 MT1: OCTNOV
```

This tape would have runs made in October and November. This step is not necessary if the user is adding runs to a tape that is already partly filled. In such a case, be careful not to initialize!

b. Mount the tape with the MOUNT command.

```
MOUNT MT1: OCTNOV
```

This is not the same as physically mounting the tape on the tape drive, which is assumed to have taken place previously.

c. Copy all the run files from disk to tape with the COPY command. For example in this case:

```
COPY *OCT.*;* MT1:
COPY *NOV.*;* MT1:
```

would copy all the October and all the November runs onto the tape mounted on tape drive 1.

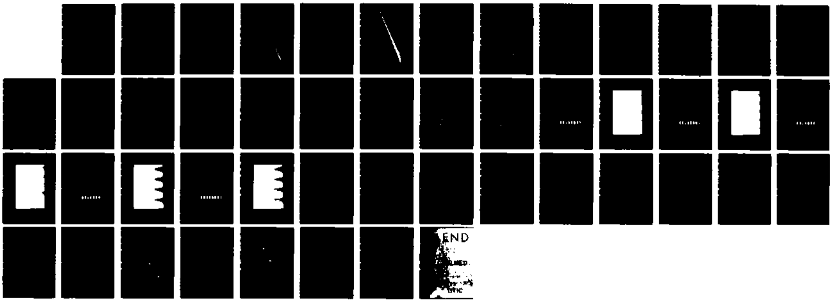
d. For future reference, it is good to get a list of names of the files on the tape. This is accomplished as follows:

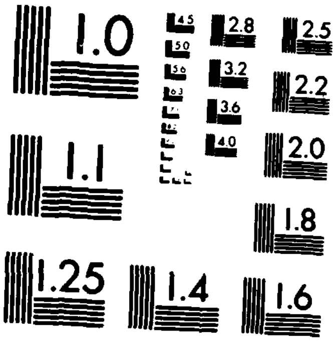
```
DIRECTORY/PRINT MT1:
```

The printout resulting from this command does not have the tape name on it, so the user should write this on immediately.
If, for some reason, this printout is lost, the tape can be remounted and the directory can be taken again. But this is no reason to be careless with the original directory printout.

AD-A147 492 ARCTIC MARINE ACOUSTICS(U) LAMONT-DOHERTY GEOLOGICAL 3/3
OBSERVATORY PALISADES NY H W KUTSCHALE OCT 84
LDGO05-20081 N00014-80-C-0021

UNCLASSIFIED F/G 20/1 NL





MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS-1963-A

e. Dismount the tape with the dismount command:

```
DISMOUNT MT1:
```

f. Delete the files on the disk that have just been copied to the tape. In this case:

```
DELETE *OCT.*;*
DELETE *NOV.*;*
```

4. Distributing the Program by Tape

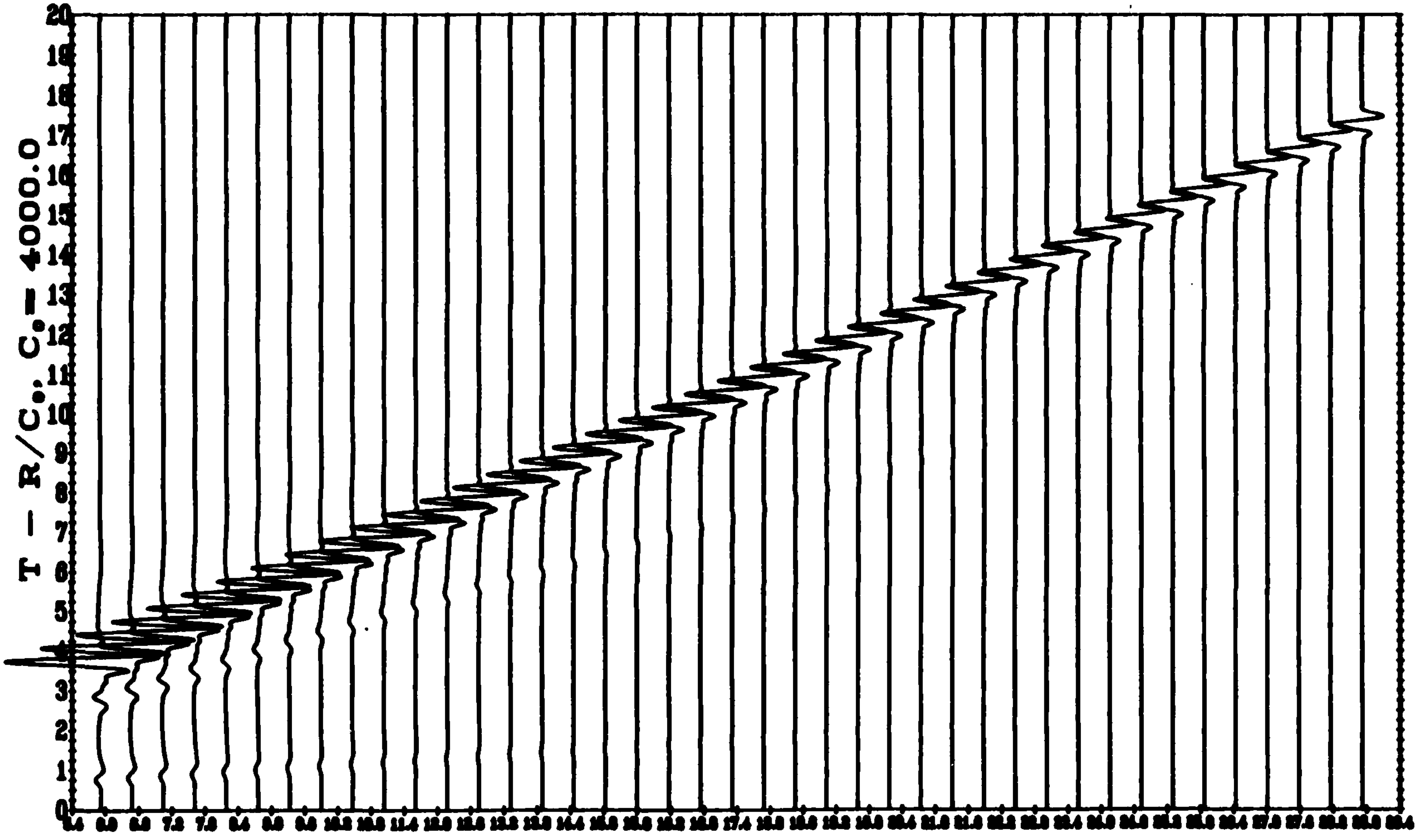
The program and test cases are distributed using tapes with the same format as the archival tapes. This is the Files-11 format, peculiar to Digital Equipment Corporation. The tapes from this system have a density of 1600 bpi, and a blocksize of 2048 bytes.

If any further questions arise, feel free to call:

Dr. H. Kutschale	(914) 359-2900 x382
Mr. L. Carroll	(914) 359-2900 x221

```
IEG03OCT.DAT;2                    6-APR-1983 15:29
                 TEST CASE DATA

06 41   0   1  1   1  0  0                  DATA FOR IMPULS11G
 0.00        1500.0               .00000150                 1.00
50.00        1500.0               .00000150                 1.00
100.0        1500.0               .00000150                 1.00
100.0        3000.0     1400.0    .00000770   .000016       2.50
125.0        3000.0     1400.0    .00000770   .000016       2.50
125.0        3000.0     1400.0    .00000770   .000016       2.50
   2     3     00 2048    00   001      1    00.50    10000.       1000.0    8000.0
0.5000000 2048    60 4.1000000 0.010        05.90        2.0          7500.0     1500.0
0.0          1500.0    .0000150   1.00         0000.0      4000.0           -1      0    1
```

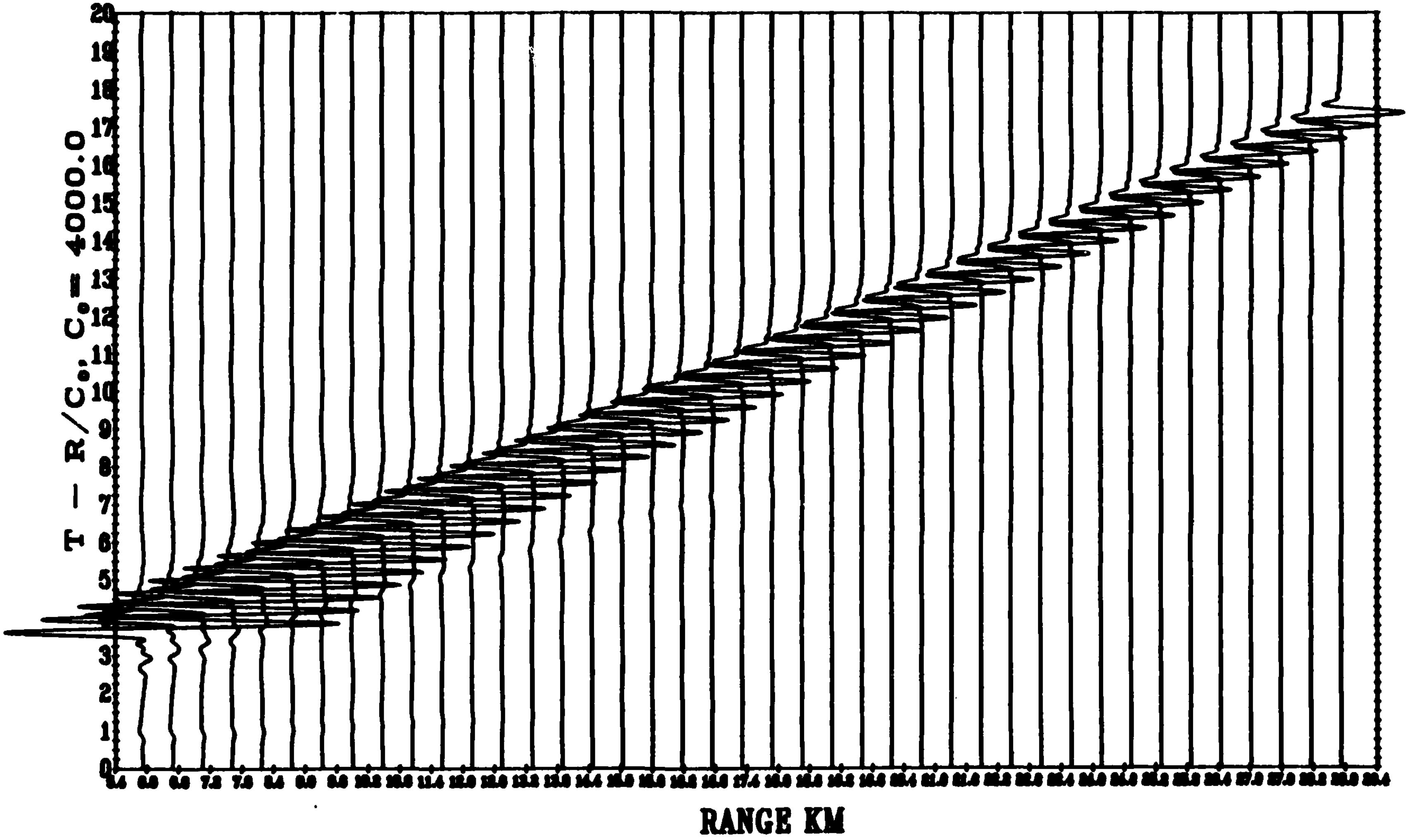
HORIZONTAL PARTICLE VELOCITY



TEST CASE DATA

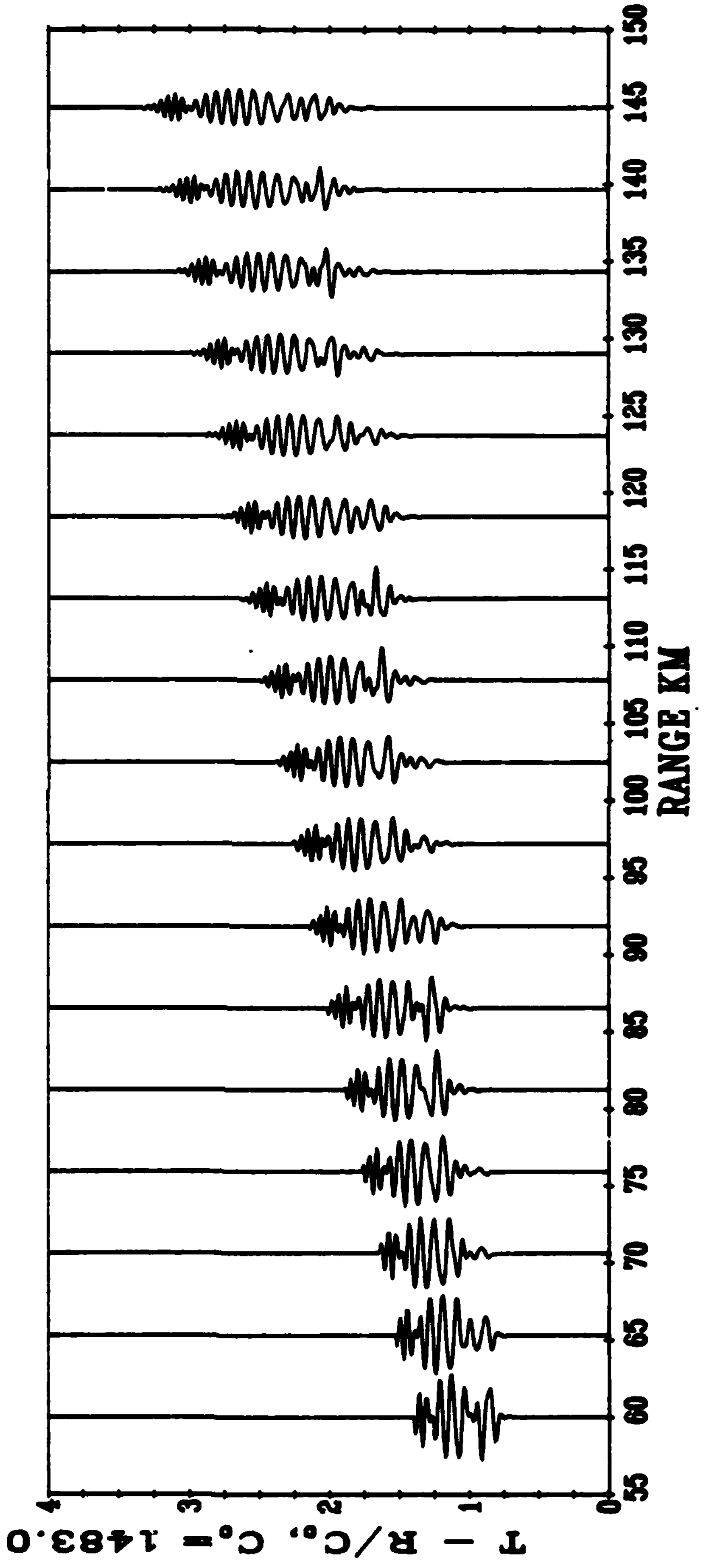
```
06 41  0  1  1  1  0  0              DATA FOR IMPULS11G
 0.00     1500.0             .00000150             1.00
50.00     1500.0             .00000150             1.00
100.0     1500.0             .00000150             1.00
100.0     3000.0    1400.0   .00000770   .000016   2.50
125.0     3000.0    1400.0   .00000770   .000016   2.50
125.0     3000.0    1400.0   .00000770   .000016   2.50
   2    3   00 2048   00   001    1   00.50   10000.     1000.0    8000.0
0.5000000 2048   60 4.1000000 0.010    05.90      2.0        7500.0    1500.0
0.0       1500.0   .0000150   1.00     0000.0    4000.0       01    0    1
```

VERTICAL PARTICLE VELOCITY



TEST CASE DATA

```
39 17  0  0  0  1  0
000.0     1434.0              .000005             1.0
050.0     1434.0              .000005             1.0
050.0     1437.0              .000005             1.0
100.0     1437.0              .000005             1.0
100.0     1440.0              .000005             1.0
150.0     1440.0              .000005             1.0
150.0     1443.0              .000005             1.0
200.0     1443.0              .000005             1.0
200.0     1446.4              .000005             1.0
250.0     1446.4              .000005             1.0
250.0     1450.0              .000005             1.0
300.0     1450.0              .000005             1.0
300.0     1453.2              .000005             1.0
350.0     1453.2              .000005             1.0
350.0     1456.0              .000005             1.0
600.0     1456.0              .000005             1.0
600.0     1460.0              .000005             1.0
800.0     1460.0              .000005             1.0
800.0     1463.0              .000005             1.0
1000.0    1463.0              .000005             1.0
1000.     1467.0              .000005             1.0
1280.     1467.0              .000005             1.0
1280.     1472.0              .000005             1.0
1500.     1472.0              .000005             1.0
1500.     1476.0              .000100             1.0
1760.     1476.0              .000100             1.0
1760.     1480.0              .000100             1.0
2000.     1480.0              .000100             1.0
2000.     1484.0              .000100             1.0
2280.     1484.0              .000100             1.0
2280.     1489.0              .000100             1.0
2600.     1489.0              .000100             1.0
2600.     1494.0              .001000             1.0
2840.     1494.0              .001000             1.0
2840.     1498.0              .001000             1.0
3040.     1498.0              .001000             1.0
3040.     1502.0              .001000             1.0
3240.     1502.0              .001000             1.0
3240.     1600.0              .010000             1.0
   5   05     0  256    0   00     1   5.0    10000.0    1433.5    1483.0
4.882813  512    40 24.414063 0.0080    60.0      8.0
0.0        0.00      00.0      00.00     000.0     1483.0      0    0   1
```


RANGE KM
55
60
65
70
75
80
85
90
95
100
105
110
115
120
125
130
135
140
145
150
0
1
2
3
4
T — R/C₀, C₀= 1483.0

```
IEG07NOV.DAT;2                   6-APR-1983 15:29
               TEST CASE DATA
05  3  0  1  1  1  1  1               DATA FOR DISPERSION
 0.00     1500.0              .0000000             1.00        0.0
 0.00     1450.0              .0000000             1.00        0.0
300.0     1450.0              .0000000             1.00        0.0
300.0     1500.0              .0000000             1.00        0.0
600.0     1500.0              .0000000             1.00        0.0
   3    3   00 1024   00  100     1   40.00   10000.    1450.1    1499.9
2.0000000 512    20 20.000000 0.0080     60.00     8.0       1450.0    1499.0
0.0       1500.0    000.0     001.00    0.00       0000.        0    1    0
```

```
DEPTH          ALPHA         BETA       ATTENA        ATTENB          RHO     ROUGHNESS

  0 00000   1500. 00000    0. 00000   0 00000000   0. 00000000     1. 00000     0. 000
  0 00000   1450. 00000    0. 00000   0 00000000   0. 00000000     1. 00000     0. 000
300 00000   1450. 00000    0. 00000   0. 00000000  0. 00000000     1. 00000     0. 000
300 00000   1500. 00000    0. 00000   0. 00000000  0. 00000000     1. 00000     0. 000
300. 00000  1500. 00000    0. 00000   0. 00000000  0. 00000000     1. 00000     0. 000
                FLUID UPPER HALF SPACE

               IMPULSE, PRESSURE, MICROBARS

                NO SURFACE ROUGHNESS


    40 00000 RSPACE
     0. 00000 DL
    40. 00000 DLRANG
            1 ISPACE
            3 NOUTPT
            3 LMAX

  KS        LR        IRES        LS          KKK       IBEAM       NM      FREQO      PO          C          C

   3         2          0        1024          0         100         1       2. 00   10000. 00   1450. 10     14

  KSI       LRI       MMAXI       MMAX        KMIN       MAX        KHPV     IDISP     KURVE      IDENOM

   3         3          5          4           2          2          0         1         0          1

  MX      FREQMAX       DT         RO         CRED

  61       20 000     0 008      50 000     1499. 900

            8. 00000 SOURCE FREQUENCY
                   8 STARTING CYCLE FOR XINPUT
                 512 TOTAL CYCLES
                  20 INIF, INTEGRAND FREQUENCY
             0. 24414 DLFREQ, DELTA FREQ.
           512. 00000 ODTDF, (1. 0 / (DT*DF))

0 3000E+03 SOURCE DEPTH
0 3000E+03 RECEIVER DEPTH

    REFLTO          ALPHAO          ATTENO          RHOO            H
 0 00000D+00     0. 15000D+04    0 00000D+00    0. 10000D+01    0. 00000D+00
```

CHART 4

PHASE VELOCITY

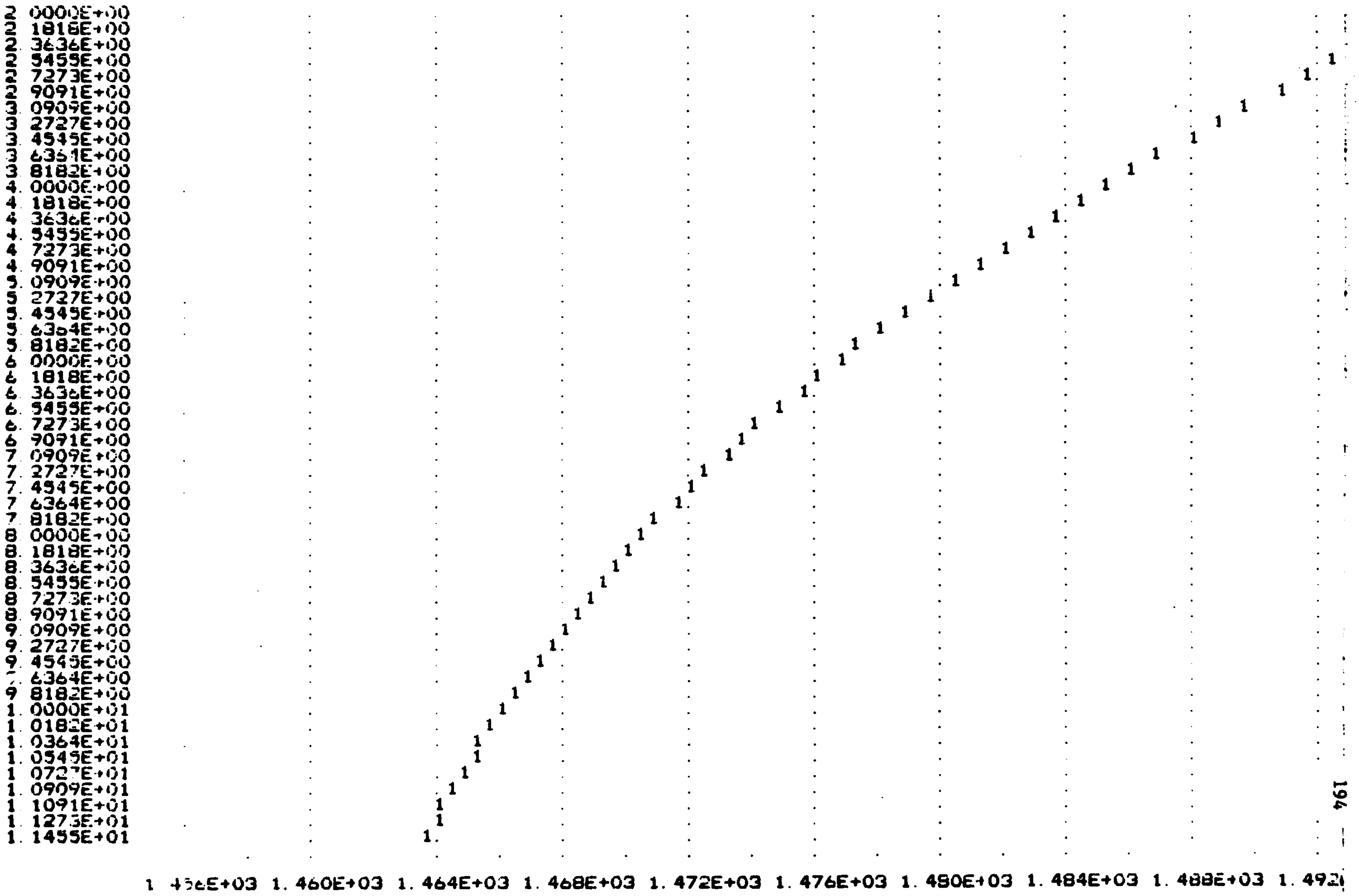
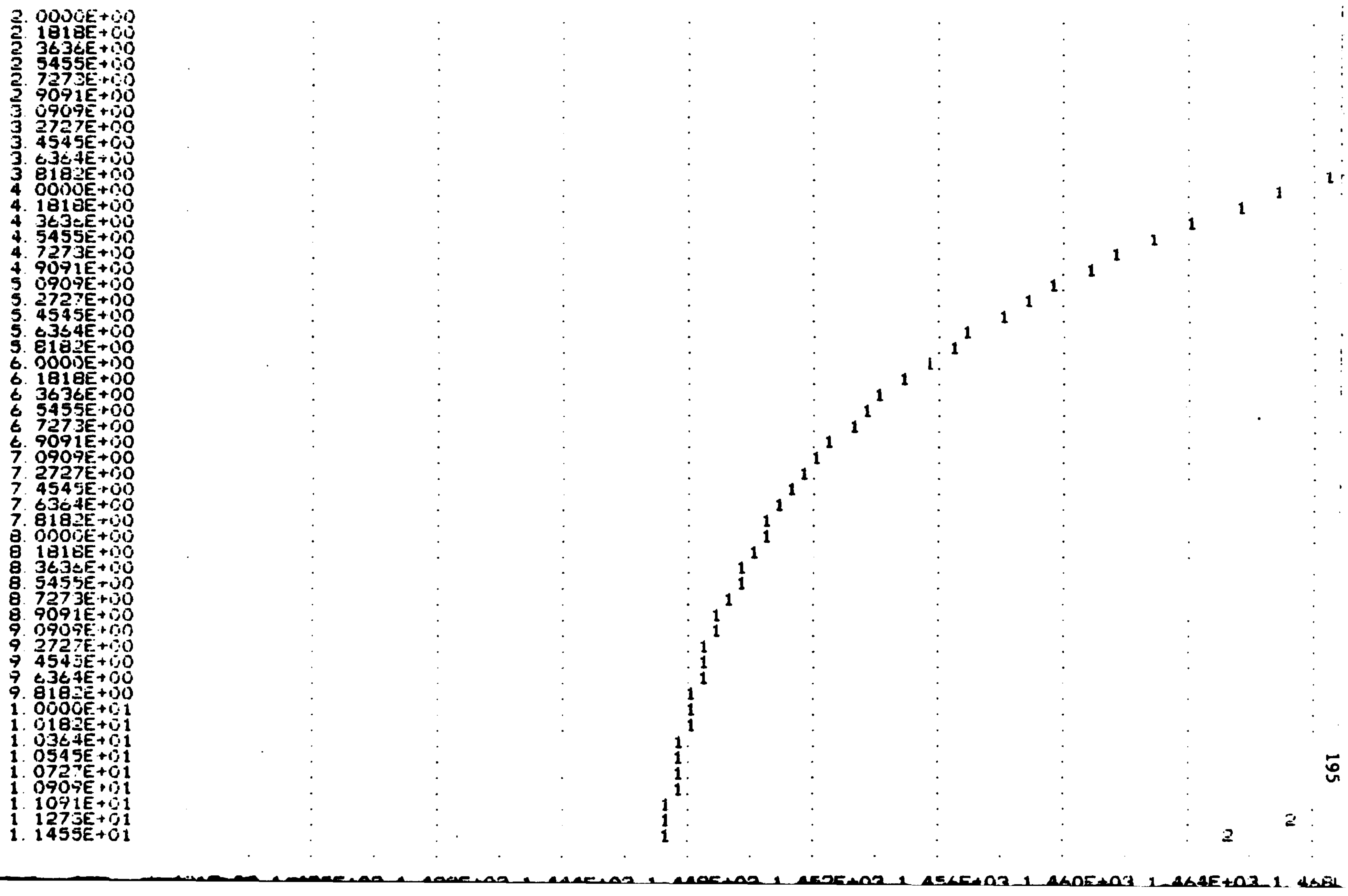


CHART 4

GROUP VELOCITY



PELOSSX USER'S MANUAL

A. Introduction

PELOSSX, our PE transmission loss program, was completed in November of 1982.

PELOSSX has these features and options for the user:

1. The PE method of computation

2. A comprehensive selection of modelling parameters. These include:
   a. A detailed sound velocity profile
   b. Choice of source depth
   c. Choice of detector depth
   d. Choice of frequency
   e. Optional ability to vary the velocity profile with range

3. Output. This includes:
   a. A printout of all input
   b. Optional contour plot of the transmission loss in a field of depth vs. range
   c. Optional page plot of transmission loss vs. range for a given detector depth

4. Numerical output of the program for further processing or plotting. Formerly, this was called punched output, since it went out to the card punch.

B. Input Variables and Formats

Program input for PELOSSX includes both the physical model and program control. This section of the manual will list and explain each line of the input.

1. Entry 1, one line

MMAX, KKK, MRMAX, NFLAG, IDUMM, LVX, DEPLR — variable names
(6I3, 2X, F10.0) — format

MMAX is a control variable which is used to read in the correct number of points in the sound velocity profile, using input type A. (see Entry 4.) If depths and velocities are input, (input type B), a non-zero value is needed here also because the depths and velocities are converted to type A input. MMAX must be a (power of 2) plus 1 because of its close association with the FFT. Maximum allowable input value is 129. (See also LVX below, the next to last item in Entry 1.)

KKK is a flag that determines whether or not this is the last case. Enter zero for the last case, or non-zero to go back to the top and go through the program again.

MRMAX is the number of sound velocity profiles. This is used for sound velocity profile range dependency. Maximum value is 16.

NFLAG is a flag that directs the program to plot a single page plot of transmission loss and/or a contour plot of transmission loss. A negative value will yield only a contour plot, a zero value will yield both types of plot, and a positive value will yield a single page (one detector depth) plot only.

IDUMM is a dummy variable at present.

LVX is both a flag and a counter value. If zero, the sound velocity profile, (Entry 4, below) consists of eight sound velocity values per line; MMAX values per profile. This is input type A. If LVX is greater than zero, the sound velocity profile consists of 'LVX' lines of depth and velocity values. This is input type B.

DEPLR is the receiver or detector depth, in meters.

2. Entry 2, one line

N, NSTEP, FREQ, ZMAX, SNORM, FGW, DRKM, GD — variable names
(8F10.0) — format

N is the number of points in the refined velocity profile. Velocity values in Entry 5 are interpolated to yield N velocity values. Also, N must be a power of 2, since it is the number of FFT points. Maximum value is 512.

NSTEP is the number of computational steps taken in range. NSTEP times DRKM (see below) gives the total range of the plot.

FREQ is the input frequency, in Hertz.

ZMAX is the maximum depth, in meters.

SNORM is the reducing velocity, in meters per sec.

FGW is an initialization factor which is normally set to 1.0

DRKM is the length of the spacing between computational points, in kilometers.

GD is the source depth, in meters.

3. Entry 3, from one to three lines
(RPTS(I), I = 1, MRMAX) variable name
(8F10.2) format

RPTS is an array that contains range points. These range points specify the ranges at which the program switches from one velocity profile to the next. The last range point of every run should be out further than the maximum range (i.e. NSTEP*DRKM). A run with only one velocity profile still needs a range point entered.

4. Entry 4, Input type A, MMAX entries (max. 17 lines, 129 values)
(C(I), I = 1,MMAX) variable name
(8F10.0) format

C is an array that has velocity values in it. These velocity values are assumed to be at equally spaced intervals along the velocity profile starting at the top (the surface of the water) and ending at ZMAX.

Input type B, LVX lines (max. 87 lines, 87 points)
((V(M,N), N=1,2), M=1, LVX) variable name
(2F10.4) format

V is an array that contains the sound velocity profile. For each point in the profile, and on each line, (one point per line), the program expects:
a. a depth value, in meters
b. a sound (pressure) velocity value, in meters per second

Input type B is flagged by a positive value of LVX.
Entry 4 is repeated MRMAX number of times.

C. Operating Procedures

The primary computer at Lamont-Doherty Geological Observatory is a VAX 11/780, which runs under the VMS operating system. Listed below are four distinctly different operations that are used to process PELOSSX.

1. Compiling-Linking

   a. Compiling (FORT command)

   Compiling the PELOSSX program on the VAX 11/780 has always been done with the following switches on:

   1. /CHECK Forbids array pointer overflow.
   2. /NOI4 Integer variables are set to 2 bytes.
   3. /NOF77 Where the compiler needs to choose between Fortran 77 and Fortran 4+, the latter is chosen.
   4. /LIST A compiler listing is made and stored on the disk as PELOSSX.LIS.

   b. Linking (LINK command)

   Linking is straight-forward. No extra routines are linked with PELOSSX.

2. Running

   PELOSSX is normally run through a batch stream, in particular the SLOW batch stream, since this is the cheapest route, especially at night. However, there is another batch stream with higher priority (and higher cost) that can be used for quicker results, the SYS$BATCH batch stream. In order to use either of these two batch streams:

   a. The executable file (PELOSSX.EXE) must be ready to use. The LINK operation satisfies this requirement.

   b. The data file must also be ready to use. This is normally prepared by the user using the editor, (and the PELOSSX USER'S MANUAL, if necessary) at a terminal.

   c. Submit the command file, PELOSSX.COM, to the batch stream (or batch queue) using a command such as one of these:

```
SUBMIT/NAME=PEL07NOV/QUEUE=SLOW/AFTER=20:00 PELOSSX
      SLOW queue, Job starts at 8:00 P.M.

SUBMIT/NAME=PEL07NOV/QUEUE=SLOW PELOSSX
      SLOW queue, Job starts now

SUBMIT/NAME=PEL07NOV PELOSSX
      SYS$BATCH queue ( the 'regular' queue ),
      Job starts now
```

   d. The following command file, also prepared using the editor, is a typical file submitted, using one of the previous submit commands.

```
$ASSIGN PEL07NOV SYS$PRINT                   (1)
$ASSIGN PEL07NOV SYS$OUTPUT                  (2)
$ASSIGN DR0:PEL07NOV.PLT FOR007              (3)
$ASSIGN DR0:PEL07NOV.OUT FOR006              (4)
$ASSIGN DR0:STANDARD.DPE FOR005              (5)
$TYPE PELOSSX                                (6)
$COPY DR0:STANDARD.DPE DR0:PEL07NOV.DAT      (7)
$RUN PELOSSX                                 (8)
```

This run is called the PEL07NOV run. 'PEL' stands for PARABOLIC EQUATION LOSS. '07NOV' indicates that this is the 7th run in November.

The first two lines in this command file put the log file on the disk instead of the printer.
The third line puts unit 7 output into file PELO7NOV.PLT.
The fourth line puts unit 6 output into file PELO7NOV.OUT.
The fifth line draws unit 5 input from STANDARD.DPE.
The sixth line copies PELOSSX.LIS to PELO7NOV.LIS.
The seventh line copies STANDARD.DPE to PELO7NOV.DAT.
The eighth line runs program PELOSSX.

A command file such as this yields a file group; each file having the same name but a different extension. These files are then treated collectively. They are printed together and stored on tape together. Such a group is, for example:

PELO7NOV.DAT	a copy of the data file
PELO7NOV.LIS	the program listing file
PELO7NOV.LOG	the log file
PELO7NOV.OUT	the output file
PELO7NOV.PLT	the numerical output file

The user normally gets a complete printout of the run by typing the command:

```
PRINT PELO7NOV.*;*
```

on the terminal.

3. Copying Runs to Tape (Archiving)

Periodically, it is necessary to clear the old runs off the disk and put them on tape. This is a very simple operation and proceeds as follows:

a. Initialize a tape. Give it a name that is somehow indicative of the present month. For example:

```
INITIALIZE/DENSITY=1600 MT1: OCTNOV
```

This tape would have runs made in October and November. This step is not necessary if the user is adding runs to a tape that is already partly filled. In such a case, be careful not to initialize!

b. Mount the tape with the MOUNT command.

```
MOUNT MT1: OCTNOV
```

This is not the same as physically mounting the tape on the tape drive, which is assumed to have taken place previously.

c. Copy all the run files from disk to tape with the COPY command. For example in this case:

```
COPY *OCT.*;* MT1:
COPY *NOV.*;* MT1:
```

would copy all the October and all the November runs onto the tape mounted on tape drive 1.

d. For future reference, it is good to get a list of names of the files on the tape. This is accomplished as follows:

```
DIRECTORY/PRINT MT1:
```

The printout resulting from this command does not have the tape name on it, so the user should write this on immediately.
If, for some reason, this printout is lost, the tape can be remounted and the directory can be taken again. But this is no reason to be careless with the original directory printout.

e. Dismount the tape with the dismount command:

```
DISMOUNT MT1:
```

f. Delete the files on the disk that have just been copied to the tape. In this case:

```
DELETE *OCT.*;*
DELETE *NOV.*;*
```

4. Distributing the Program by Tape

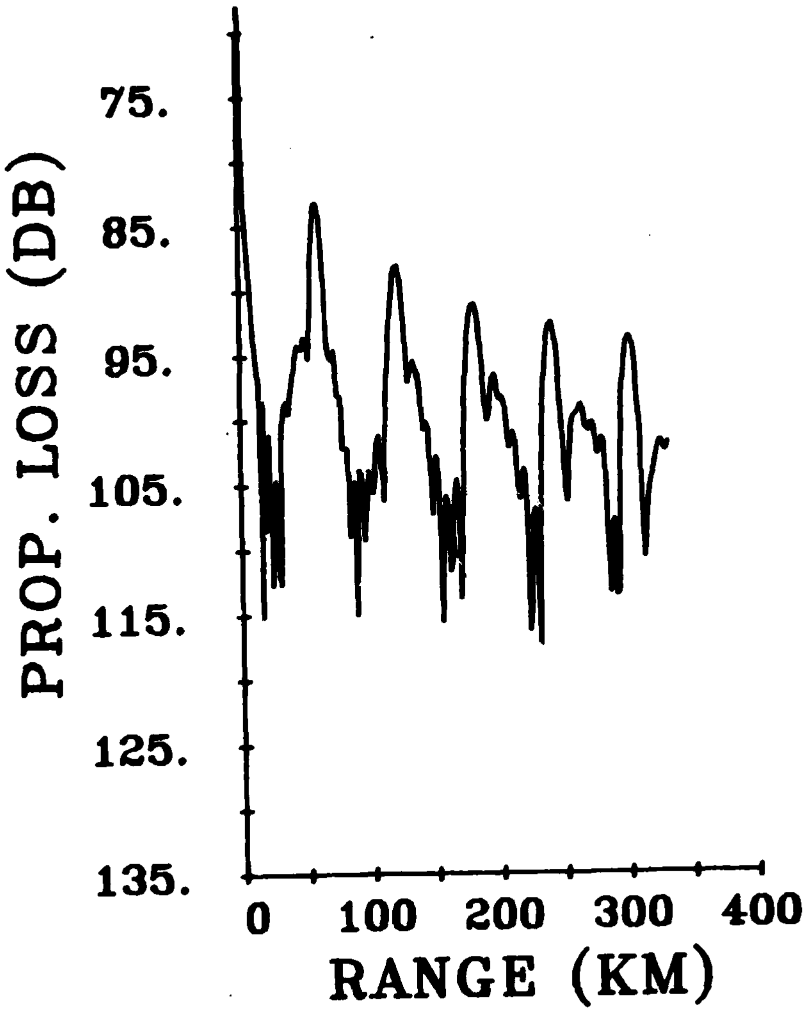
The program and test cases are distributed using tapes with the same format as the archival tapes. This is the Files-11 format, peculiar to Digital Equipment Corporation. The tapes from this system have a density of 1600 bpi, and a blocksize of 2048 bytes.

If any further questions arise, feel free to call:

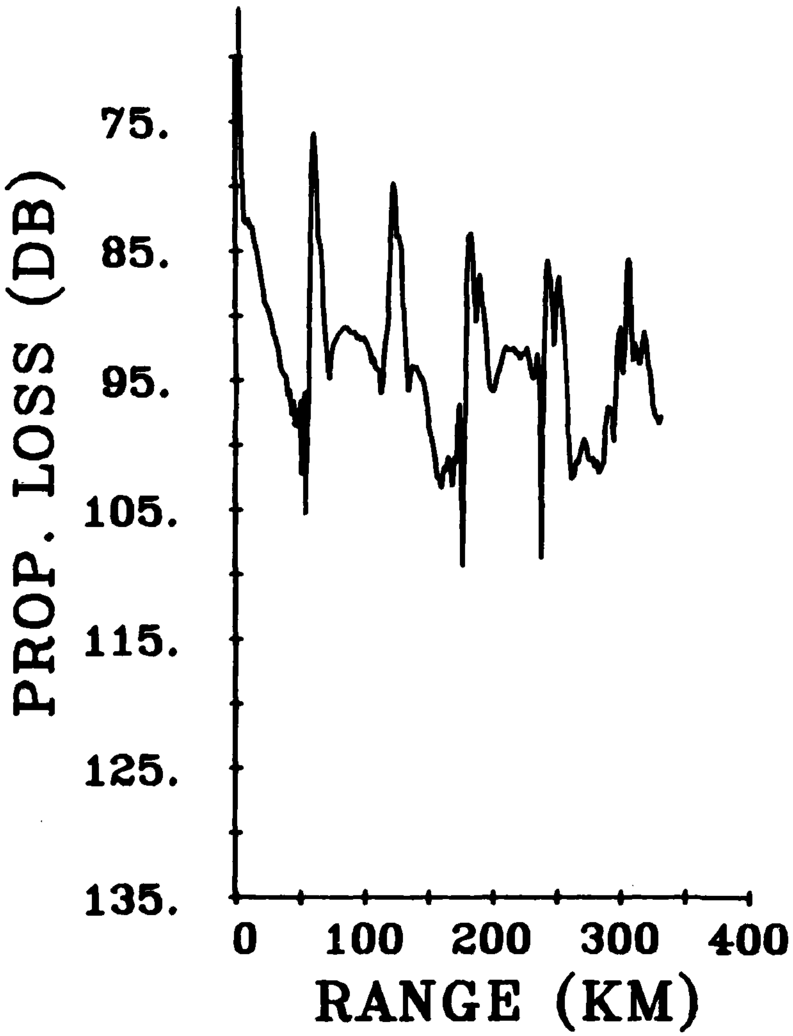
Dr. H. Kutschale	(914) 359-2900 x382
Mr. L. Carroll	(914) 359-2900 x221

```
                    TEST CASE DATA
                                                        DATA FOR PELOSSX
129   1   1   0   0  12   91.4
          1024          200  025.0     3500.0     1499.0     1.0        1.6616    61.0
 999.0
 00.0       1494.0     0.000      0.00001    0.00000    1.03
 61.0       1494.7     0.000      0.00001    0.00000    1.03
 91.4       1495.2     0.C00      0.00001    0.00000    1.03
 330.0      1496.8     0.000      0.00001    0.00000    1.03
 800.0      1487.9     0.000      0.00001    0.00000    1.03
 1950.0     1495.6     0.000      0.00001    0.00000    1.03
 3400.0     1511.7     0.000      0.00001    0.00000    1.03
 3400.0     1511.7     0.000      0.00001    0.00000    1.03
 3450.0     1511.7     0.000      0.00001    0.00000    1.03
 3450.0     1511.7     0.000      0.00001    0.00000    1.03
 3500.0     1511.7     0.000      0.00001    0.00000    1.03
 3500.0     1511.7     0.000      0.00001    0.00000    1.03
129   1   1   0   0  12   91.4
          1024          200  050.0     3500.0     1499.0     1.0        1.6616    61.0
 999.0
 00.0       1494.0     0.000      0.00001    0.00000    1.03
 61.0       1494.7     0.000      0.00001    0.00000    1.03
 91.4       1495.2     0.000      0.00001    0.00000    1.03
 330.0      1496.8     0.000      0.00001    0.00000    1.03
 800.0      1487.9     0.000      0.00001    0.00000    1.03
 1950.0     1495.6     0.000      0.00001    0.00000    1.03
 3400.0     1511.7     0.000      0.00001    0.00000    1.03
 3400.0     1511.7     0.000      0.00001    0.00000    1.03
 3450.0     1511.7     0.000      0.00001    0.00000    1.03
 3450.0     1511.7     0.000      0.00001    0.00000    1.03
 3500.0     1511.7     0.000      0.00001    0.00000    1.03
 3500.0     1511.7     0.000      0.00001    0.00000    1.03
129   1   1   0   0  12   91.4
          1024          200  075.0     3500.0     1499.0     1.0        1.6616    61.0
 999.0
 00.0       1494.0     0.000      0.00001    0.00000    1.03
 61.0       1494.7     0.000      0.00001    0.00000    1.03
 91.4       1495.2     0.000      0.00001    0.00000    1.03
 330.0      1496.8     0.000      0.00001    0.00000    1.03
 800.0      1487.9     0.000      0.00001    0.00000    1.03
 1950.0     1495.6     0.000      0.00001    0.00000    1.03
 3400.0     1511.7     0.000      0.00001    0.00000    1.03
 3400.0     1511.7     0.000      0.00001    0.00000    1.03
 3450.0     1511.7     0.000      0.00001    0.00000    1.03
 3450.0     1511.7     0.000      0.00001    0.00000    1.03
 3500.0     1511.7     0.000      0.00001    0.00000    1.03
 3500.0     1511.7     0.000      0.00001    0.00000    1.03
129   1   1   0   0  12   91.4
          1024          400  120.0     3500.0     1499.0     1.0        0.6923    61.0
 999.0
 00.0       1494.0     0.000      0.00001    0.00000    1.03
 61.0       1494.7     0.000      0.00001    0.00000    1.03
 91.4       1495.2     0.000      0.00001    0.00000    1.03
 330.0      1496.8     0.000      0.00001    0.00000    1.03
 800.0      1487.9     0.000      0.00001    0.00000    1.03
 1950.0     1495.6     0.000      0.00001    0.00000    1.03
 3400.0     1511.7     0.000      0.00001    0.00000    1.03
 3400.0     1511.7     0.000      0.00001    0.00000    1.03
 3450.0     1511.7     0.000      0.00001    0.00000    1.03
 3450.0     1511.7     0.000      0.00001    0.00000    1.03
 3500.0     1511.7     0.000      0.00001    0.00000    1.03
 3500.0     1511.7     0.000      0.00001    0.00000    1.03
129   0   1   0   0  12   91.4
          1024          400  160.0     3500.0     1499.0     1.0        0.6923    61.0
 999.0
 00.0       1494.0     0.000      0.00001    0.00000    1.03
 61.0       1494.7     0.000      0.00001    0.00000    1.03
 91.4       1495.2     0.000      0.00001    0.00000    1.03
 330.0      1496.8     0.000      0.00001    0.00000    1.03
 800.0      1487.9     0.000      0.00001    0.00000    1.03
 1950.0     1495.6     0.000      0.00001    0.00000    1.03
 3400.0     1511.7     0.000      0.00001    0.00000    1.03
 3400.0     1511.7     0.000      0.00001    0.00000    1.03
 3450.0     1511.7     0.000      0.00001    0.00000    1.03
 3450.0     1511.7     0.000      0.00001    0.00000    1.03
 3500.0     1511.7     0.000      0.00001    0.00000    1.03
 3500.0     1511.7     0.000      0.00001    0.00000    1.03
```

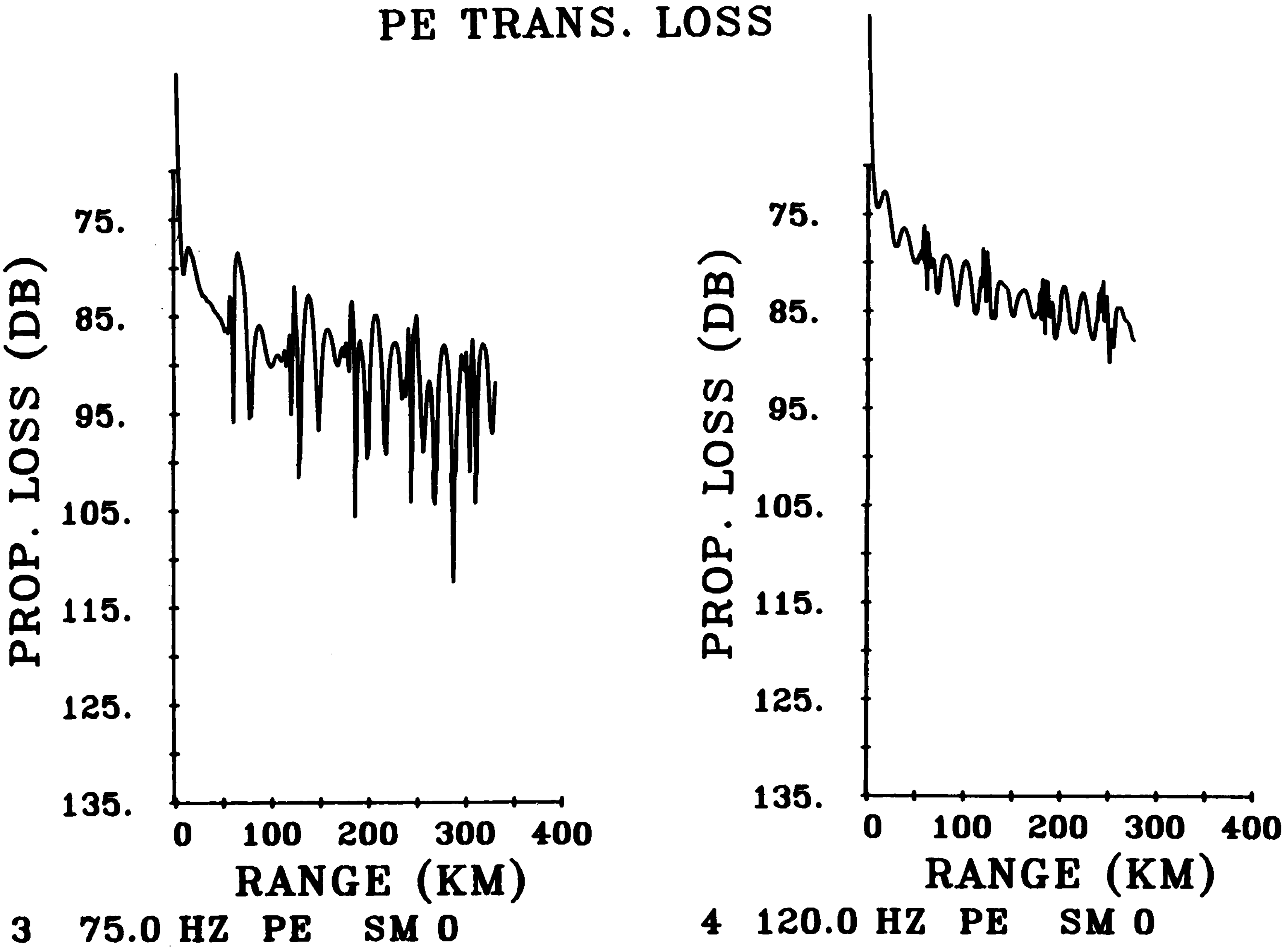
PE TRANS. LOSS

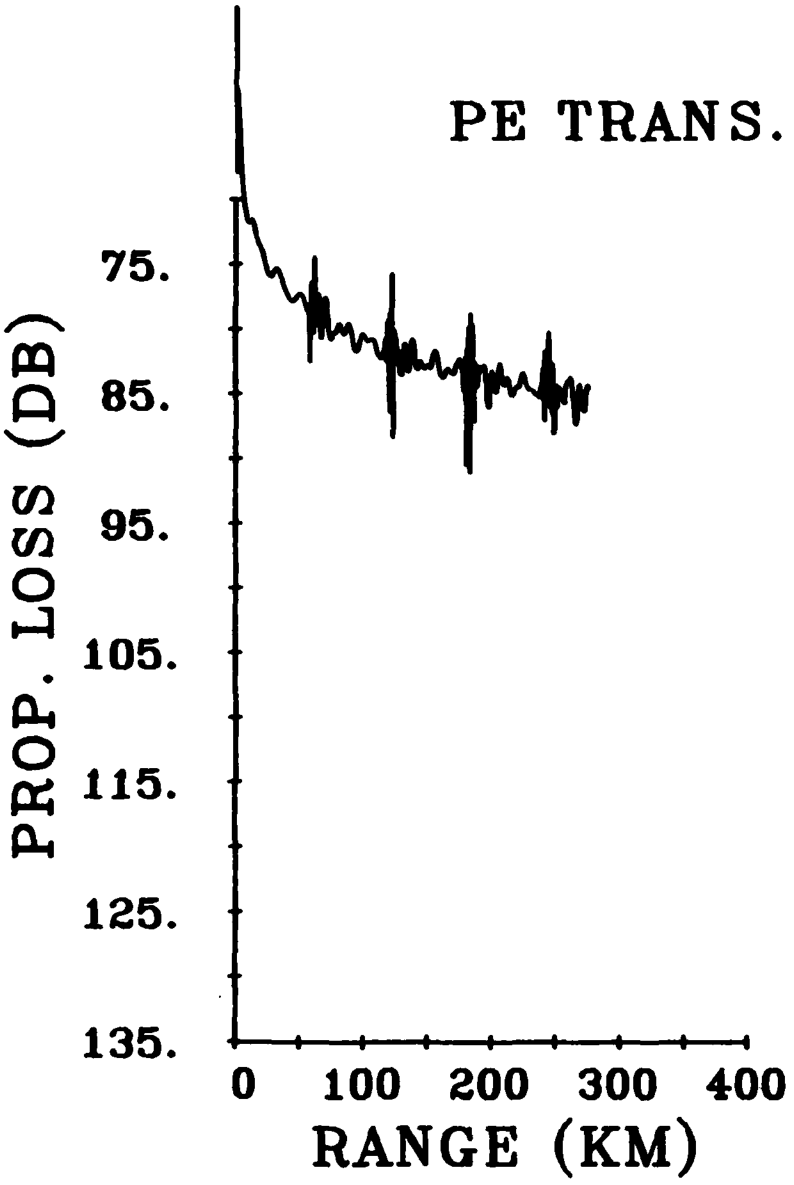


1 25.0 HZ PE SM 0



2 50.0 HZ PE SM 0


PE TRANS. LOSS
PROP. LOSS (DB)
75.
85.
95.
105.
115.
125.
135.
0
100
200
300
400
RANGE (KM)
3 75.0 HZ PE SM 0
PROP. LOSS (DB)
75.
85.
95.
105.
115.
125.
135.
0
100
200
300
400
RANGE (KM)
4 120.0 HZ PE SM 0



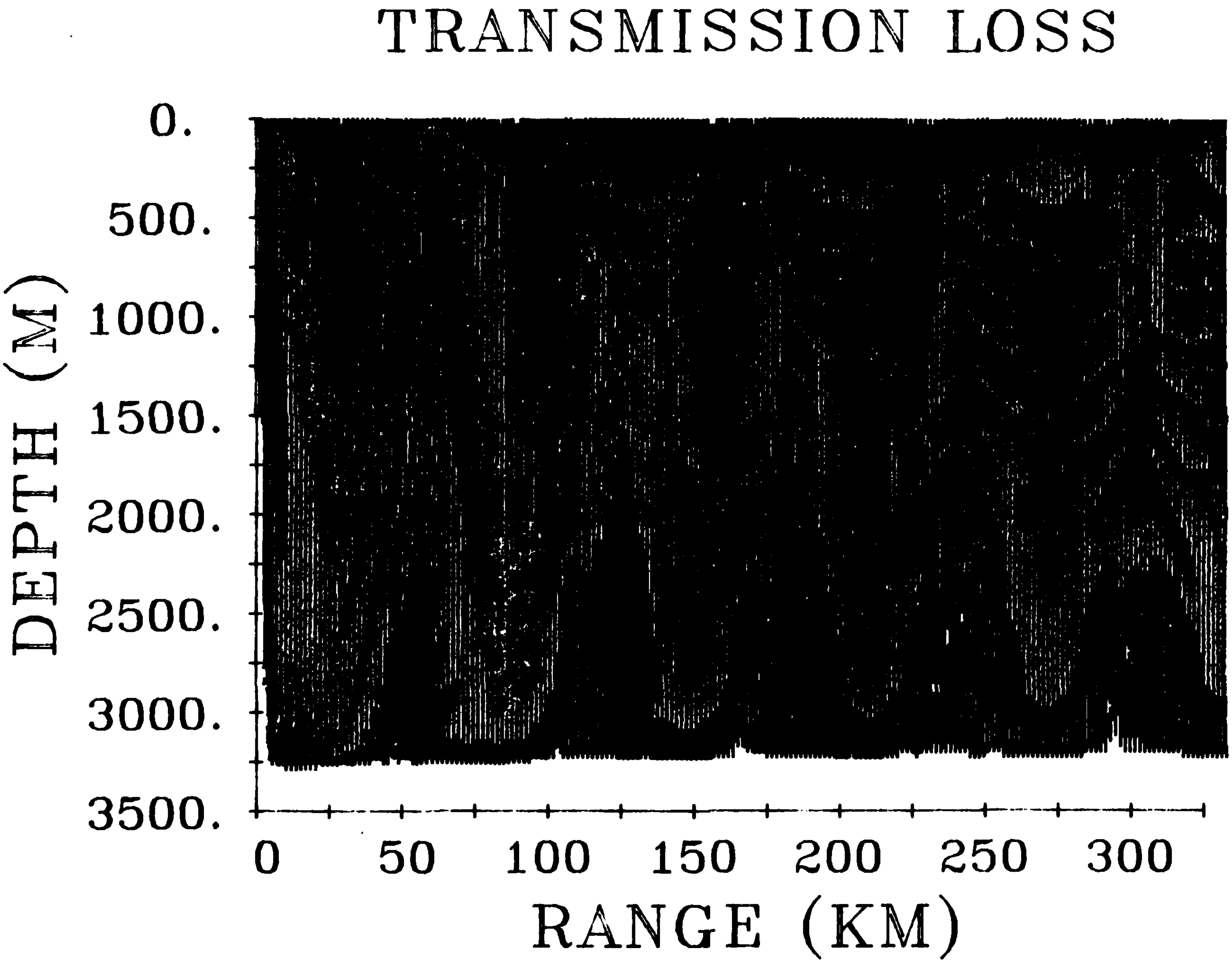
5 160.0 HZ PE SM 0

25 HZ

COLOR to DB LOSS

CALIBRATION

1	■	< 77.924
2	■	77.924 to 85.873
3	▨	85.873 to 93.822
4	▨	93.822 to 101.772
5	■	101.772 to 109.721
6	■	109.721 to 117.671
7	■	117.671 to 125.620
8	▥	125.620 to 133.569
9	□	>133.569


TRANSMISSION LOSS
DEPTH (M)
0.
500.
1000.
1500.
2000.
2500.
3000.
3500.
0
50
100
150
200
250
300
RANGE (KM)

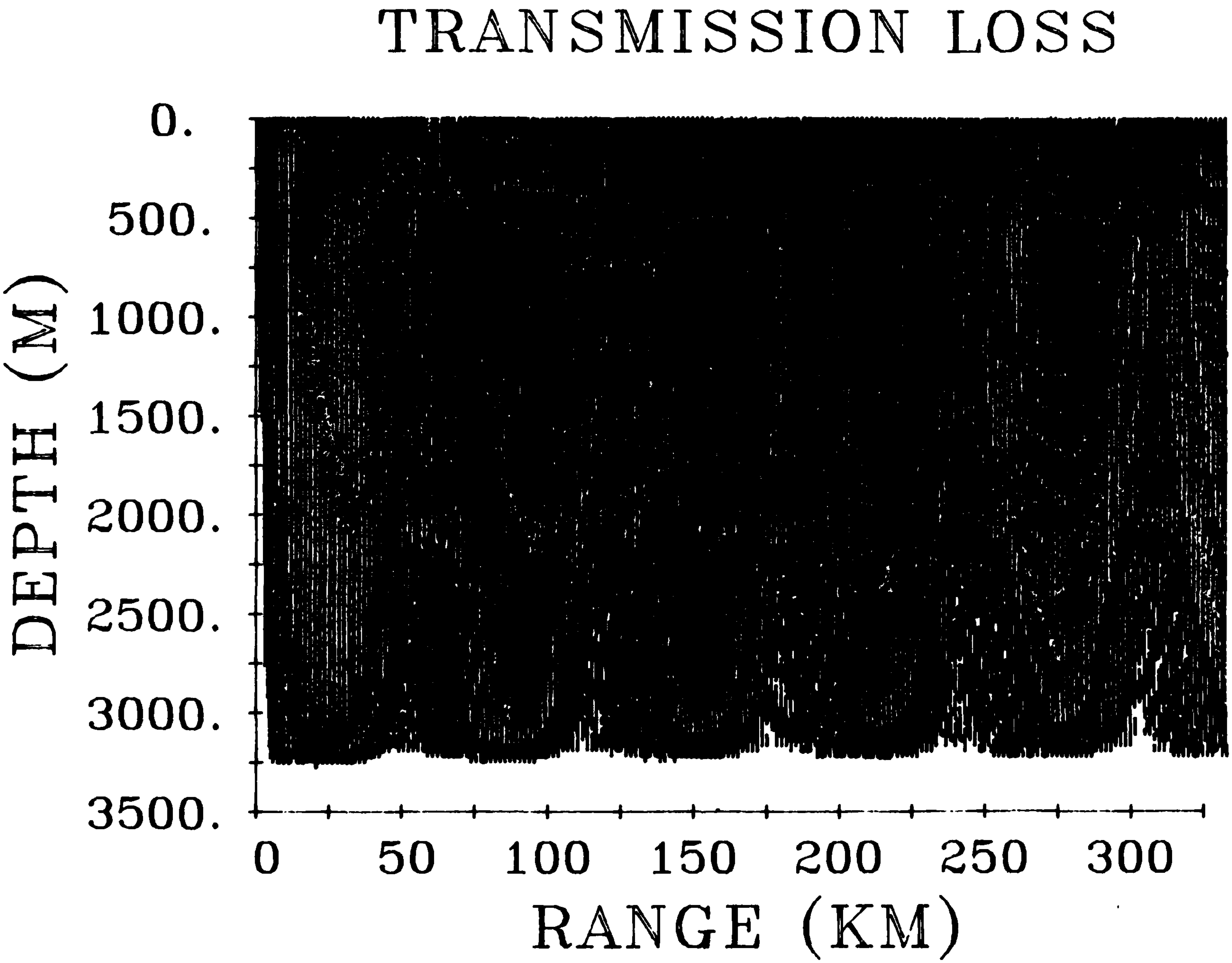
25HZ

50 HZ

COLOR to DB LOSS

CALIBRATION

1	■	< 75.706
2	■	75.706 to 83.727
3	■	83.727 to 91.748
4	■	91.748 to 99.769
5	■	99.769 to 107.789
6	■	107.789 to 115.810
7	■	115.810 to 123.831
8	▥	123.831 to 131.852
9	□	>131.852



TRANSMISSION LOSS

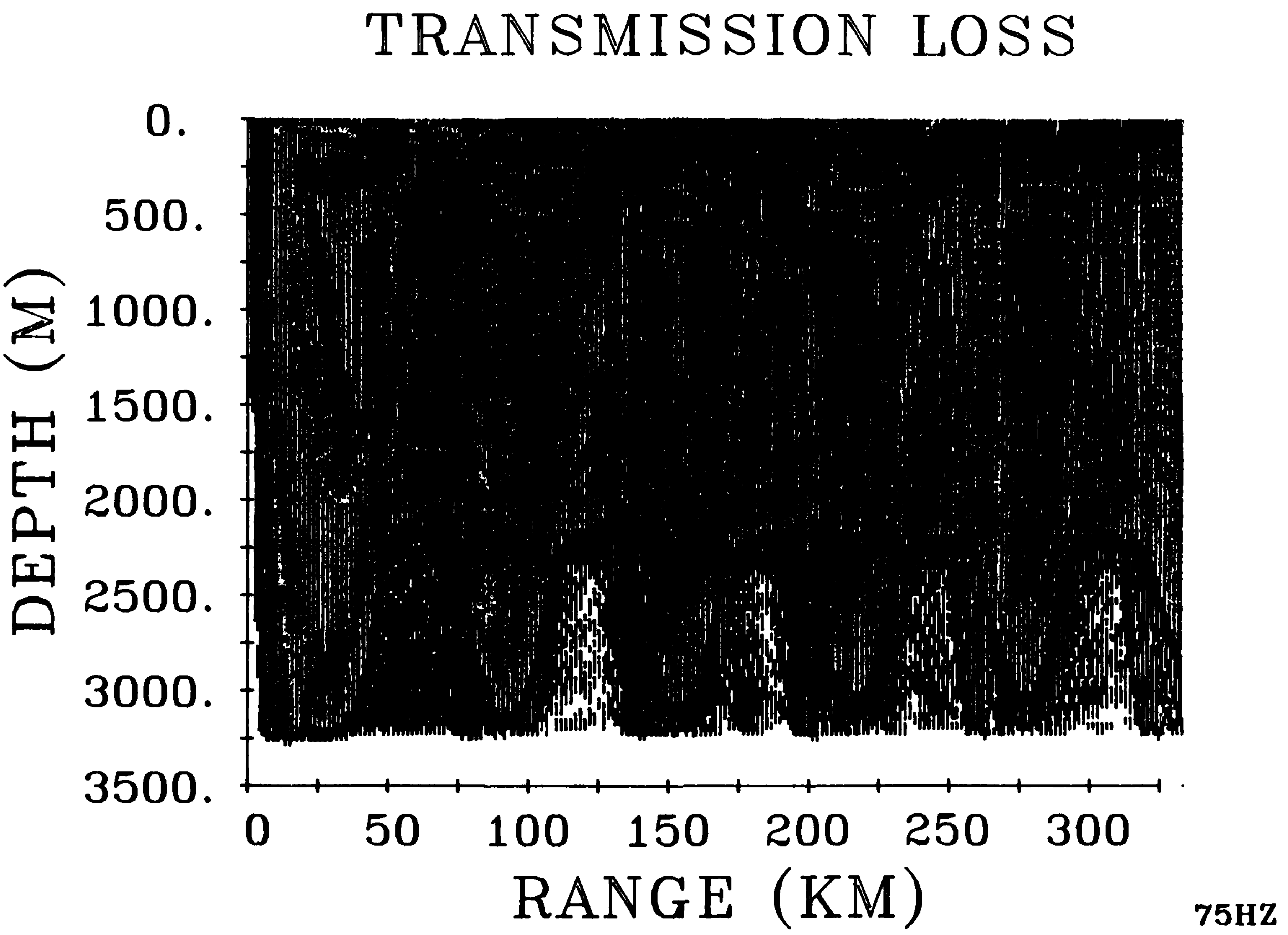
50HZ

75 HZ

COLOR to DB LOSS

CALIBRATION

1	■	< 75.138
2	■	75.138 to 83.161
3	▩	83.161 to 91.184
4	▩	91.184 to 99.206
5	■	99.206 to 107.229
6	■	107.229 to 115.252
7	■	115.252 to 123.274
8	▥	123.274 to 131.297
9	□	>131.297

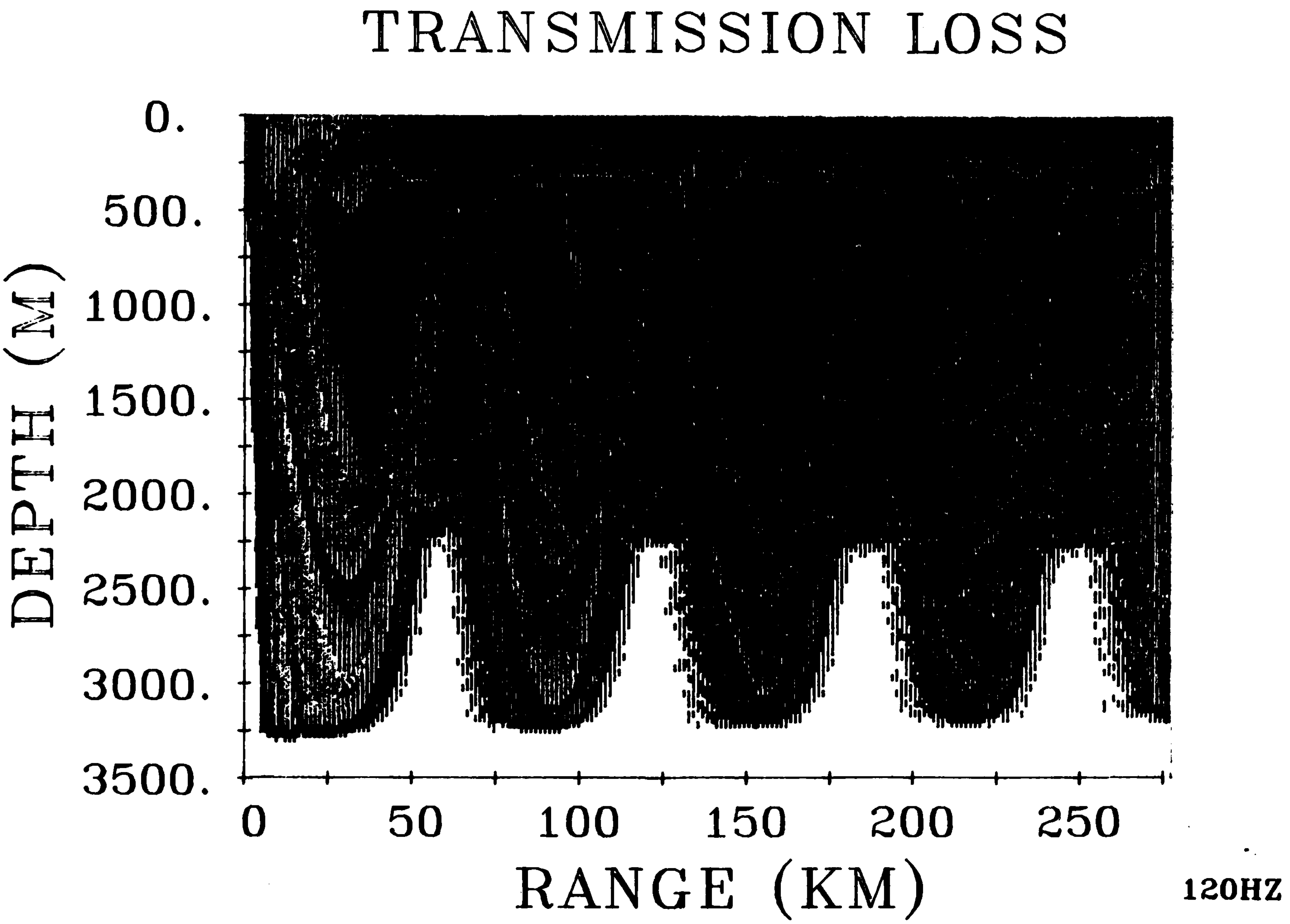

TRANSMISSION LOSS
DEPTH (M)
0.
500.
1000.
1500.
2000.
2500.
3000.
3500.
0
50
100
150
200
250
300
RANGE (KM)
75HZ

120 HZ

COLOR to DB LOSS

CALIBRATION

1	■	< 69.128
2	■	69.128 to 77.308
3	▩	77.308 to 85.489
4	■	85.489 to 93.670
5	■	93.670 to 101.851
6	■	101.851 to 110.032
7	■	110.032 to 118.213
8	▥	118.213 to 126.394
9	□	>126.394

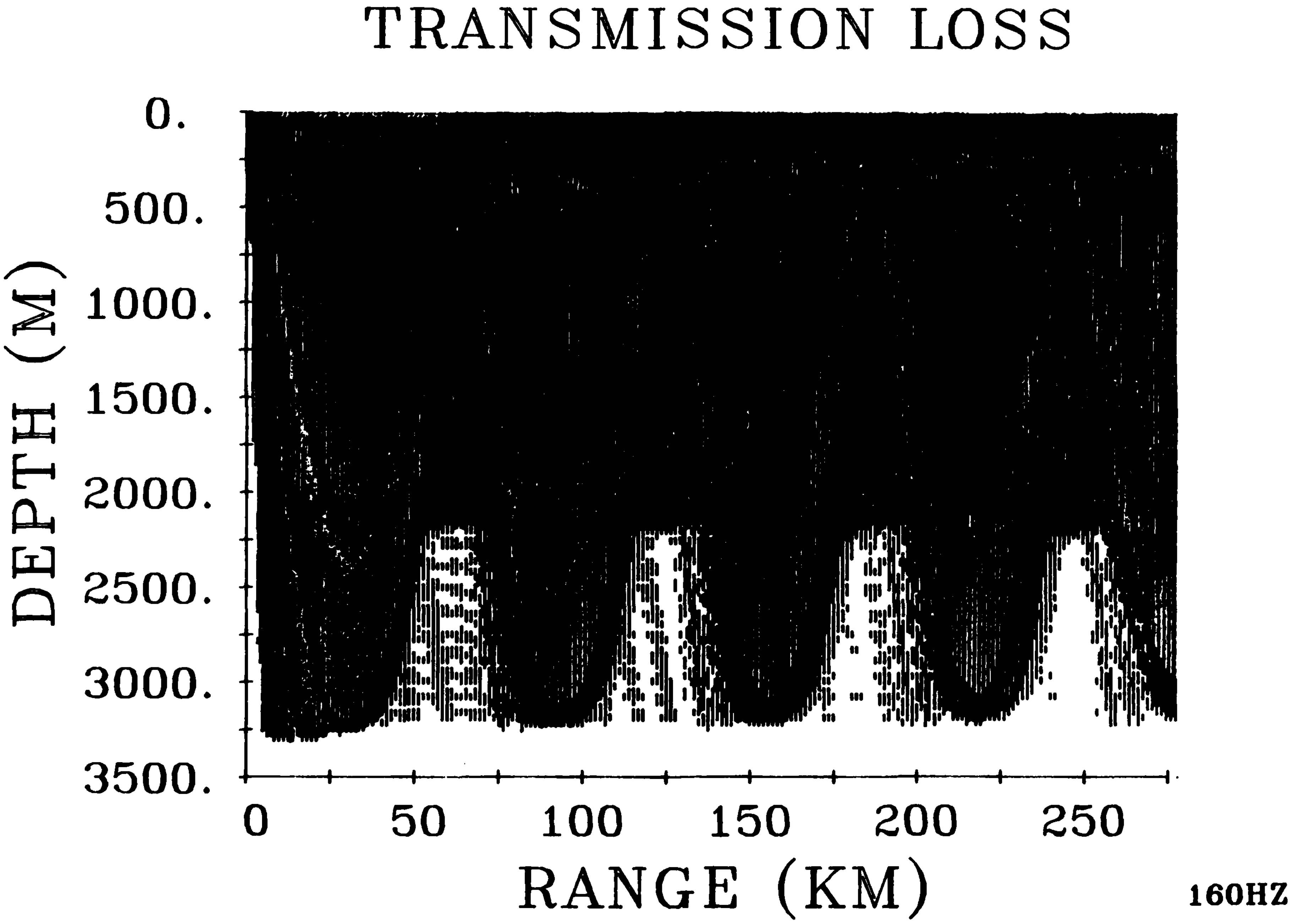

TRANSMISSION LOSS
DEPTH (M)
0.
500.
1000.
1500.
2000.
2500.
3000.
3500.
0
50
100
150
200
250
RANGE (KM)
120HZ

160 HZ

COLOR to DB LOSS

CALIBRATION

1	■	< 71.396
2	■	71.396 to 79.517
3	▨	79.517 to 87.639
4	▨	87.639 to 95.760
5	■	95.760 to 103.881
6	■	103.881 to 112.002
7	■	112.002 to 120.123
8	▥	120.123 to 128.244
9	□	>128.244


TRANSMISSION LOSS
DEPTH (M)
0.
500.
1000.
1500.
2000.
2500.
3000.
3500.
0
50
100
150
200
250
RANGE (KM)
160HZ

PEPULSEX USER'S MANUAL

A. Introduction

PEPULSEX, our PE impulse program, was completed in October of 1982.

PEPULSEX has these features and options for the user:

1. The PE method of computation

2. A comprehensive selection of modelling parameters. These include:
   - a. A detailed sound velocity profile
   - b. Choice of source depth
   - c. Choice of detector depth
   - d. Choice of frequency band and source frequency
   - e. Choice of initial range, range spacing, and range count
   - f. Optional ability to vary the velocity profile with range

3. Output. This includes:
   - a. A printout of all input This includes a printer plot of all the velocity profiles.
   - b. The pulse signal plots

4. Numerical output of the program for further processing or plotting. Formerly, this was called punched output, since it went out to the card punch.

B. Input Variables and Formats

Program input for PEPULSEX includes both the physical model and program control. This section of the manual will list and explain each line of the input.

1. Entry 1, one line

```
MMAX, KKK, MRMAX, N, LDUMM, NTIME, NOUTPT, NREVRS, NCARD, NPLOT, LV
                                                         variable name
  ( 11I3 )                                               format
```

MMAX is a control variable which is used to read in the correct number of points in the sound velocity profile, using input type A. (see Entry 5.) If depths and velocities are input, (input type B), a non-zero value is needed here also because the depths and velocities are converted to type A input. MMAX must be a (power of 2) plus 1 because of its close association with the FFT. Maximum allowable input value is 129. (See also LVX below, the last item in Entry 1.)

KKK is a flag that determines whether or not this is the last case. Enter zero for the last case, or non-zero to go back to the top and go through the program again.

MRMAX is the number of sound velocity profiles. This is used for sound velocity profile range dependency. Maximum value is 16.

N is the number of points in the refined velocity profile. Velocity values in Entry 5 are interpolated to yield N velocity values. Also, N must be a power of 2, since it is the number of FFT points at one location in the program. Maximum value is 512.

LDUMM is a dummy variable at present.

NTIME is a flag that, when set non-zero, moves the signal tim window up by the length of the window. In essence, the reducing velocity marks the end of the pulse signal plot instead of the beginning, which is the normal case.

NOUTPT is the number of output signal plots.

NREVRS is both a flag and a pointer. If non-zero, it signals that there will be pulse reversal, and it points to the output signal which is reversed and used as a source function.

NCARD is a control variable that, when set non-zero, allows the program to put out signal data on unit 7 in a form which is readable as input by another program. A zero value will prevent any output on unit 7. This is normally used to make graphic plots. (Originally, on the IBM 360/91, unit 7 was the punched card unit, hence the name NCARD.)

NPLOT is a control variable that, when set non-zero, allows the program to put out printer plots of the output signals.

LVX is both a flag and a counter value. If zero, the sound velocity profile, (Entry 5, below) consists of eight sound velocity values per line; MMAX values per profile. This is input type A. If LVX is greater than zero, the sound velocity profile consists of 'LVX' lines of depth and velocity values. This is input type B.

2. Entry 2, one line

```
RO, RSPACE, DEPLR, ZMAX, SNORM, FGW, DRKM, GD
                                                         variable name
  ( 8F10.0 )                                             format
```

RO is the initial range or the range of the first output signal, in kilometers.

RSPACE is the range spacing between output signals, in kilometers. Unless this is a multiple of DRKM, (see below) the program can only approximate this value.

DEPLR is the receiver or detector depth, in meters.

ZMAX is the maximum depth, in meters.

SNORM is the reducing velocity, in meters per sec. For each output signal, the starting time is offset from TO at range 0 (point of origin) by a time equal to that range divided by SNORM.

FGW is an initialization factor which is normally set to 1.0

DRKM is the length of the spacing between computational points, in kilometers.

GD is the source depth, in meters.

3. Entry 3, from one to three lines

```
UMIN, UMAX, ( RPTS(I), I = 1, MRMAX )          variable names
( 8F10.0 )                                     format
```

UMAX is a sound velocity (meters/sec.) that is used to specify the lower bound of the output signal.

UMIN is similar in function to UMAX, but used to specify the upper bound.

RPTS is an array that contains range points. These range points specify the ranges at which the program switches from one velocity profile to the next. The last range point of every run should be out further than the last output pulse signal. A run with only one velocity profile still needs a range point entered.

4. Entry 4, one line

```
FREGO, NX, DT, FREGS, PO, FREGM
                                               variable names
( F10.0, I5, 5X, 4F10.2 )                      format
```

FREGO is the lower bound of the frequency integration, in Hertz.

NX is the number of FFT points and the number of points in each output signal.

DT is the time difference between the successive points in the output signals, in sec.

FREGS is the sound source frequency, in Hertz.

PO is the source level. For this program we normally use a value of 10000.

FREGM is the upper bound of the frequency integration, in Hertz.

5. Entry 5, Input type A, MMAX entries (max. 17 lines, 129 values)

```
( C(I), I = 1,MMAX )                           variable name
( 8F10.0 )                                     format
```

C is an array that has velocity values in it. These velocity values are assumed to be at equally spaced intervals along the velocity profile starting at the top (the surface of the water) and ending at ZMAX.

Input type B, LVX lines (max. 87 lines, 87 points)

```
(( V(M,N), N=1,2 ), M=1, LVX )                 variable name
( 2F10.4 )                                     format
```

V is an array that contains the sound velocity profile. For each point in the profile, and on each line, (one point per line), the program expects:

a. a depth value, in meters
b. a sound (pressure) velocity value, in meters per second

Input type B is flagged by a positive value of LVX.
Entry 5 is repeated MRMAX number of times.

PEPULSEX USER'S MANUAL

C. Operating Procedures

The primary computer at Lamont-Doherty Geological Observatory is a VAX 11/780, which runs under the VMS operating system. Listed below are four distinctly different operations that are used to process PEPULSEX.

1. Compiling-Linking

   a. Compiling (FORT command)

   Compiling the PEPULSEX program on the VAX 11/780 has always been done with the following switches on:

   1. /CHECK Forbids array pointer overflow.
   2. /NOI4 Integer variables are set to 2 bytes.
   3. /NOF77 Where the compiler needs to choose between Fortran 77 and Fortran 4+, the latter is chosen.
   4. /LIST A compiler listing is made and stored on the disk as PEPULSEX.LIS.

   b. Linking (LINK command)

   Linking is straight-forward. No extra routines are linked with PEPULSEX.

2. Running

   PEPULSEX is normally run through a batch stream, in particular the SLOW batch stream, since this is the cheapest route, especially at night. However, there is another batch stream with higher priority (and higher cost) that can be used for quicker results, the SYS$BATCH batch stream. In order to use either of these two batch streams:

   a. The executable file (PEPULSEX.EXE) must be ready to use. The LINK operation satisfies this requirement.

   b. The data file must also be ready to use. This is normally prepared by the user using the editor, (and the PEPULSEX USER'S MANUAL, if necessary) at a terminal.

   c. Submit the command file, PEPULSEX.COM, to the batch stream (or batch queue) using a command such as one of these:

```
SUBMIT/NAME=PEPO5NOV/QUEUE=SLOW/AFTER=20:00 PEPULSEX
      SLOW queue, Job starts at 8:00 P.M.

SUBMIT/NAME=PEPO5NOV/QUEUE=SLOW PEPULSEX
      SLOW queue, Job starts now

SUBMIT/NAME=PEPO5NOV PEPULSEX
      SYS$BATCH queue ( the 'regular' queue ),
      Job starts now
```

   d. The following command file, also prepared using the editor, is a typical file submitted, using one of the previous submit commands.

```
$ASSIGN PEPO5NOV SYS$PRINT                    (1)
$ASSIGN PEPO5NOV SYS$OUTPUT                   (2)
$ASSIGN DRO:PEPO5NOV.PLT FOROO7               (3)
$ASSIGN DRO:PEPO5NOV.OUT FOROO6               (4)
$ASSIGN DRO:STANDARD.DP2 FOROO5               (5)
$TYPE PEPULSEX                                (6)
$COPY DRO:STANDARD.DP2 DRO:PEPO5NOV.DAT       (7)
$RUN PEPULSEX                                 (8)
```

This run is called the PEPO5NOV run. 'PEP' stands for PARABOLIC EQUATION PULSE. 'O5NOV' indicates that this is the 5th run in November.

The first two lines in this command file put the log file on the disk instead of the printer.
The third line puts unit 7 output into file PEPO5NOV.PLT.
The fourth line puts unit 6 output into file PEPO5NOV.OUT.
The fifth line draws unit 5 input from STANDARD.DP2.
The sixth line copies PEPULSEX.LIS to PEPO5NOV.LIS.
The seventh line copies STANDARD.DP2 to PEPO5NOV.DAT.
The eighth line runs program PEPULSEX.

A command file such as this yields a file group; each file having the same name but a different extension. These files are then treated collectively. They are printed together and stored on tape together. Such a group is, for example:

PEPO5NOV.DAT	a copy of the data file
PEPO5NOV.LIS	the program listing file
PEPO5NOV.LOG	the log file
PEPO5NOV.OUT	the output file
PEPO5NOV.PLT	the numerical output file

The user normally gets a complete printout of the run by typing the command:

```
PRINT PEPO5NOV.*;*
```

on the terminal.

3. Copying Runs to Tape (Archiving)

Periodically, it is necessary to clear the old runs off the disk and put them on tape. This is a very simple operation and proceeds as follows:

a. Initialize a tape. Give it a name that is somehow indicative of the present month. For example:

```
INITIALIZE/DENSITY=1600 MT1: OCTNOV
```

This tape would have runs made in October and November. This step is not necessary if the user is adding runs to a tape that is already partly filled. In such a case, be careful not to initialize!

b. Mount the tape with the MOUNT command.

```
MOUNT MT1: OCTNOV
```

This is not the same as physically mounting the tape on the tape drive, which is assumed to have taken place previously.

c. Copy all the run files from disk to tape with the COPY command. For example in this case:

```
COPY *OCT.*;* MT1:
COPY *NOV.*;* MT1:
```

would copy all the October and all the November runs onto the tape mounted on tape drive 1.

d. For future reference, it is good to get a list of names of the files on the tape. This is accomplished as follows:

```
DIRECTORY/PRINT MT1:
```

The printout resulting from this command does not have the tape name on it, so the user should write this on immediately.
If, for some reason, this printout is lost, the tape can be remounted and the directory can be taken again. But this is no reason to be careless with the original directory printout.

e. Dismount the tape with the dismount command:

```
DISMOUNT MT1:
```

f. Delete the files on the disk that have just been copied to the tape. In this case:

```
DELETE *OCT.*;*
DELETE *NOV.*;*
```

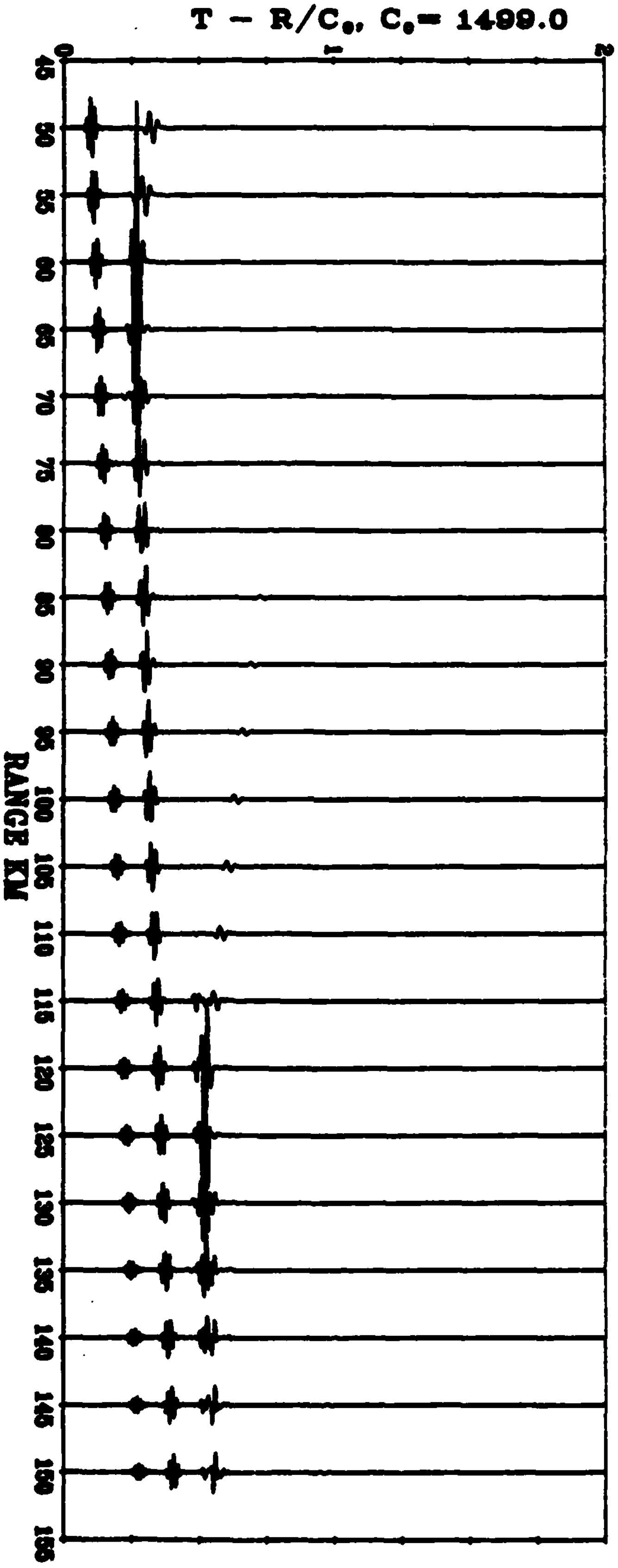
4. Distributing the Program by Tape

The program and test cases are distributed using tapes with the same format as the archival tapes. This is the Files-11 format, peculiar to Digital Equipment Corporation. The tapes from this system have a density of 1600 bpi, and a blocksize of 2048 bytes.

If any further questions arise, feel free to call:

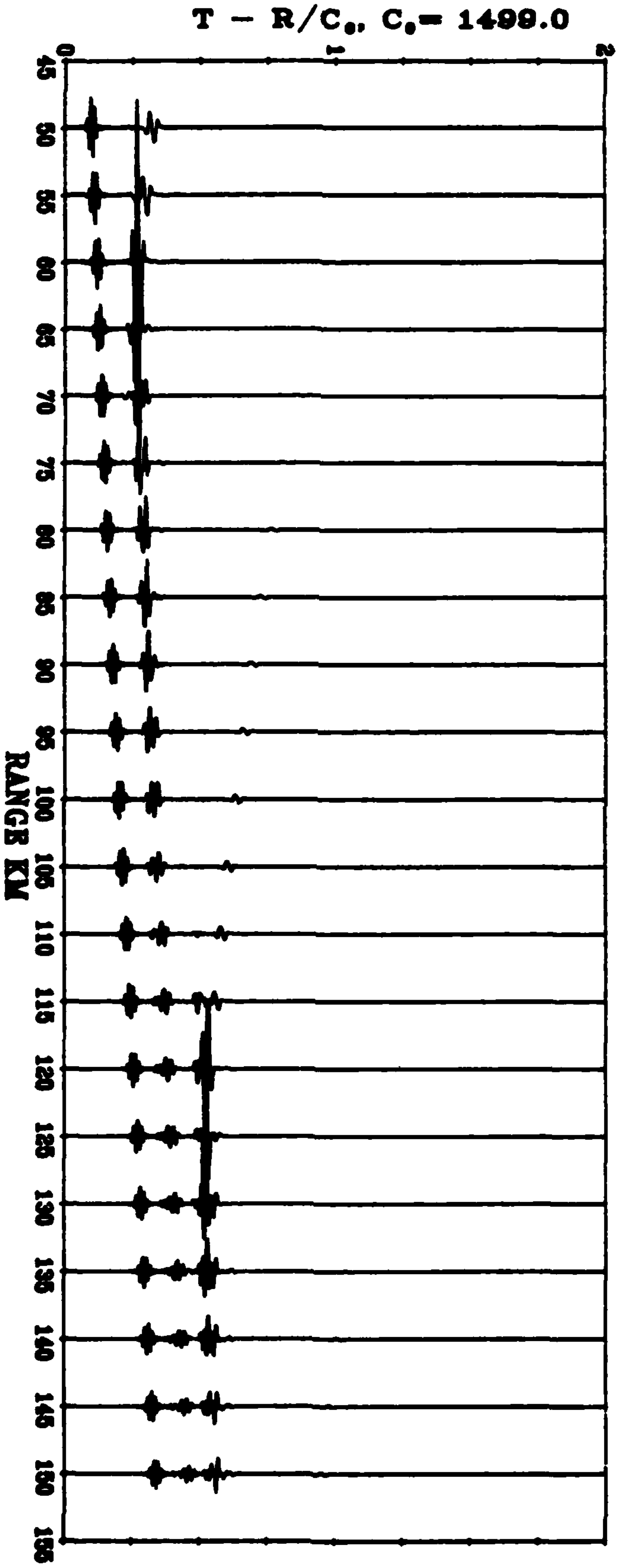
Dr. H. Kutschale	(914) 359-2900 x382
Mr. L. Carroll	(914) 359-2900 x221

```
PEPO1DEC.DAT;2                    6-APR-1983 15:32
                 TEST CASE DATA
129  0  1128 44  0 21  0  1  1 12                       DATA FOR PEPULSEX
  50.0     005.0         61.0       3500.00    1499.0    1.0       1.0       91.4
 1490.0     1505.         900.0
 12.50000  2048         0.002        50.       10000.0    102.0000
 00.0       1494.0      0.000       0.00001    0.00000   1.03
 61.0       1494.7      0.000       0.00001    0.00000   1.03
 91.4       1495.2      0.000       0.00001    0.00000   1.03
 330.0      1496.8      0.000       0.00001    0.00000   1.03
 800.0      1487.9      0.000       0.00001    0.00000   1.03
 1950.0     1495.6      0.000       0.00001    0.00000   1.03
 3400.0     1511.7      0.000       0.00001    0.00000   1.03
 3400.0     1511.7      0.000       0.00010    0.00000   1.03
 3450.0     1511.7      0.000       0.00010    0.00000   1.03
 3450.0     1511.7      0.000       0.00010    0.00000   1.03
 3500.0     1511.7      0.000       0.00100    0.00000   1.03
 3500.0     1511.7      0.000       0.00100    0.00000   1.03
```

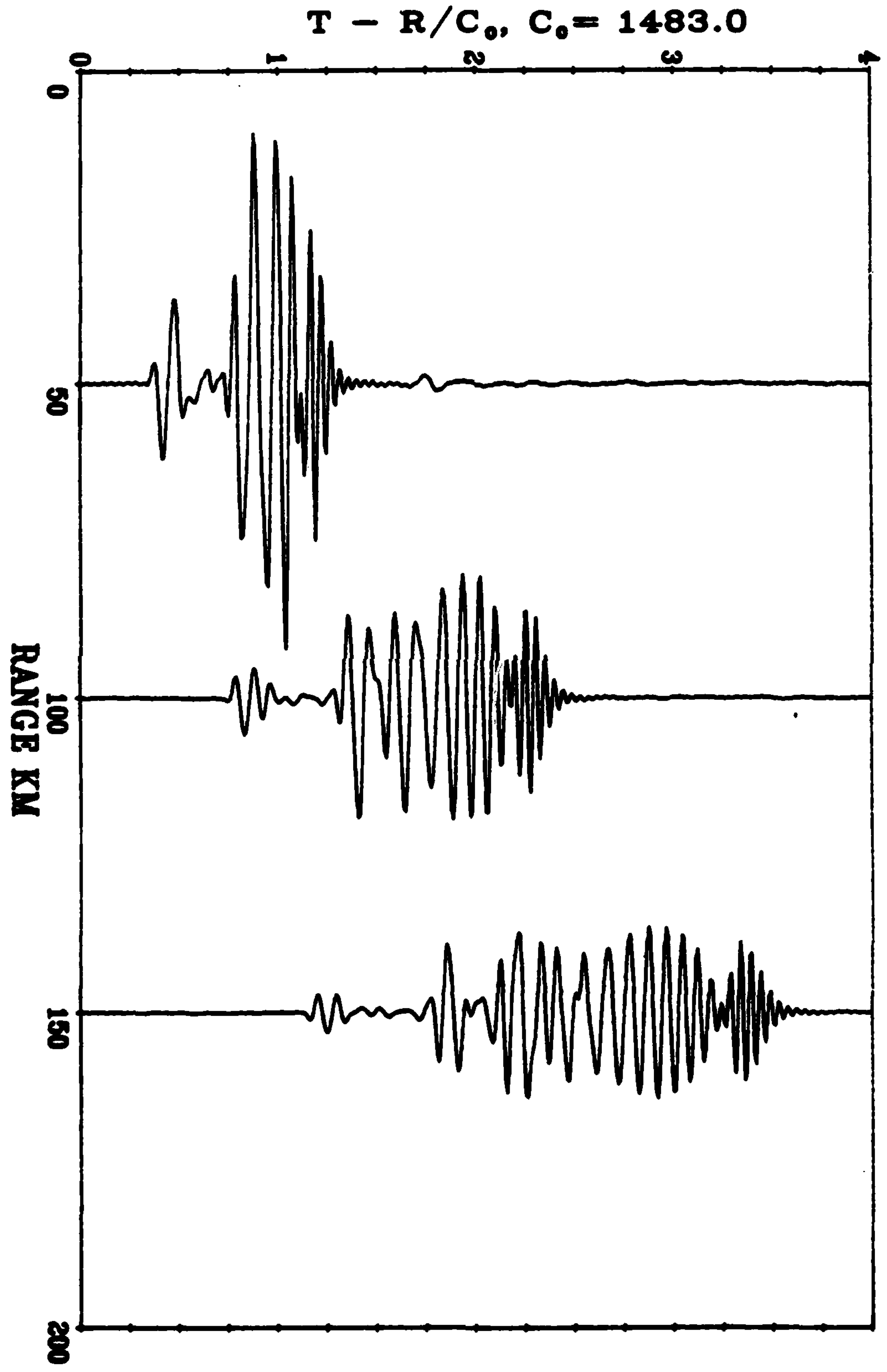

T - R/C$_0$, C$_0$= 1499.0
0
1
2
45
50
55
60
65
70
75
80
85
90
95
100
105
110
115
120
125
130
135
140
145
150
155
RANGE KM

TEST CASE DATA

```
129  0  5128 44  0 21  0  1  1 12
                                  3500.00     1499.0     1.0        1.0        91.4
  50.0       005.0         61.0
                           050.0     065.0        080.0      095.0      900.0
 1490.0      1505.
 12.50000    2048       0.002        50.          10000.0    102.0000
 00.0        1494.0      0.000       0.00001     0.00000     1.03
 27.344      1494.314    0.000       0.00001     0.00000     1.03
 61.0        1494.7      0.000       0.00001     0.00000     1.03
 82.031      1495.046    0.000       0.00001     0.00000     1.03
 91.4        1495.2      0.000       0.00001     0.00000     1.03
 109.375     1495.32     0.000       0.00001     0.00000     1.03
 136.719     1495.504    0.000       0.00001     0.00000     1.03
 330.0       1496.8      0.000       0.00001     0.00000     1.03
 800.0       1487.9      0.000       0.00001     0.00000     1.03
 1950.0      1495.6      0.000       0.00001     0.00000     1.03
 3400.0      1511.7      0.000       0.00001     0.00000     1.03
 3500.0      1511.7      0.000       0.00100     0.00000     1.03
 00.0        1494.0      0.000       0.00001     0.00000     1.03
 27.344      1494.314    0.000       0.00001     0.00000     1.03
 61.0        1494.6      0.000       0.00001     0.00000     1.03
 82.031      1494.846    0.000       0.00001     0.00000     1.03
 91.4        1494.9      0.000       0.00001     0.00000     1.03
 109.375     1495.02     0.000       0.00001     0.00000     1.03
 136.719     1495.304    0.000       0.00001     0.00000     1.03
 330.0       1496.8      0.000       0.00001     0.00000     1.03
 800.0       1487.9      0.000       0.00001     0.00000     1.03
 1950.0      1495.6      0.000       0.00001     0.00000     1.03
 3400.0      1511.7      0.000       0.00001     0.00000     1.03
 3500.0      1511.7      0.000       0.00100     0.00000     1.03
 00.0        1494.0      0.000       0.00001     0.00000     1.03
 27.344      1494.314    0.000       0.00001     0.00000     1.03
 61.0        1494.4      0.000       0.00001     0.00000     1.03
 82.031      1494.446    0.000       0.00001     0.00000     1.03
 91.4        1494.3      0.000       0.00001     0.00000     1.03
 109.375     1494.52     0.000       0.00001     0.00000     1.03
 136.719     1495.004    0.000       0.00001     0.00000     1.03
 330.0       1496.8      0.000       0.00001     0.00000     1.03
 800.0       1487.9      0.000       0.00001     0.00000     1.03
 1950.0      1495.6      0.000       0.00001     0.00000     1.03
 3400.0      1511.7      0.000       0.00001     0.00000     1.03
 3500.0      1511.7      0.000       0.00100     0.00000     1.03
 00.0        1494.0      0.000       0.00001     0.00000     1.03
 27.344      1494.314    0.000       0.00001     0.00000     1.03
 61.0        1494.1      0.000       0.00001     0.00000     1.03
 82.031      1493.846    0.000       0.00001     0.00000     1.03
 91.4        1493.4      0.000       0.00001     0.00000     1.03
 109.375     1493.82     0.000       0.00001     0.00000     1.03
 136.719     1494.604    0.000       0.00001     0.00000     1.03
 330.0       1496.8      0.000       0.00001     0.00000     1.03
 800.0       1487.9      0.000       0.00001     0.00000     1.03
 1950.0      1495.6      0.000       0.00001     0.00000     1.03
 3400.0      1511.7      0.000       0.00001     0.00000     1.03
 3500.0      1511.7      0.000       0.00100     0.00000     1.03
 00.0        1494.0      0.000       0.00001     0.00000     1.03
 27.344      1494.314    0.000       0.00001     0.00000     1.03
 61.0        1493.7      0.000       0.00001     0.00000     1.03
 82.031      1493.046    0.000       0.00001     0.00000     1.03
 91.4        1492.2      0.000       0.00001     0.00000     1.03
 109.375     1492.92     0.000       0.00001     0.00000     1.03
 136.719     1494.104    0.000       0.00001     0.00000     1.03
 330.0       1496.8      0.000       0.00001     0.00000     1.03
 800.0       1487.9      0.000       0.00001     0.00000     1.03
 1950.0      1495.6      0.000       0.00001     0.00000     1.03
 3400.0      1511.7      0.000       0.00001     0.00000     1.03
 3500.0      1511.7      0.000       0.00100     0.00000     1.03
```

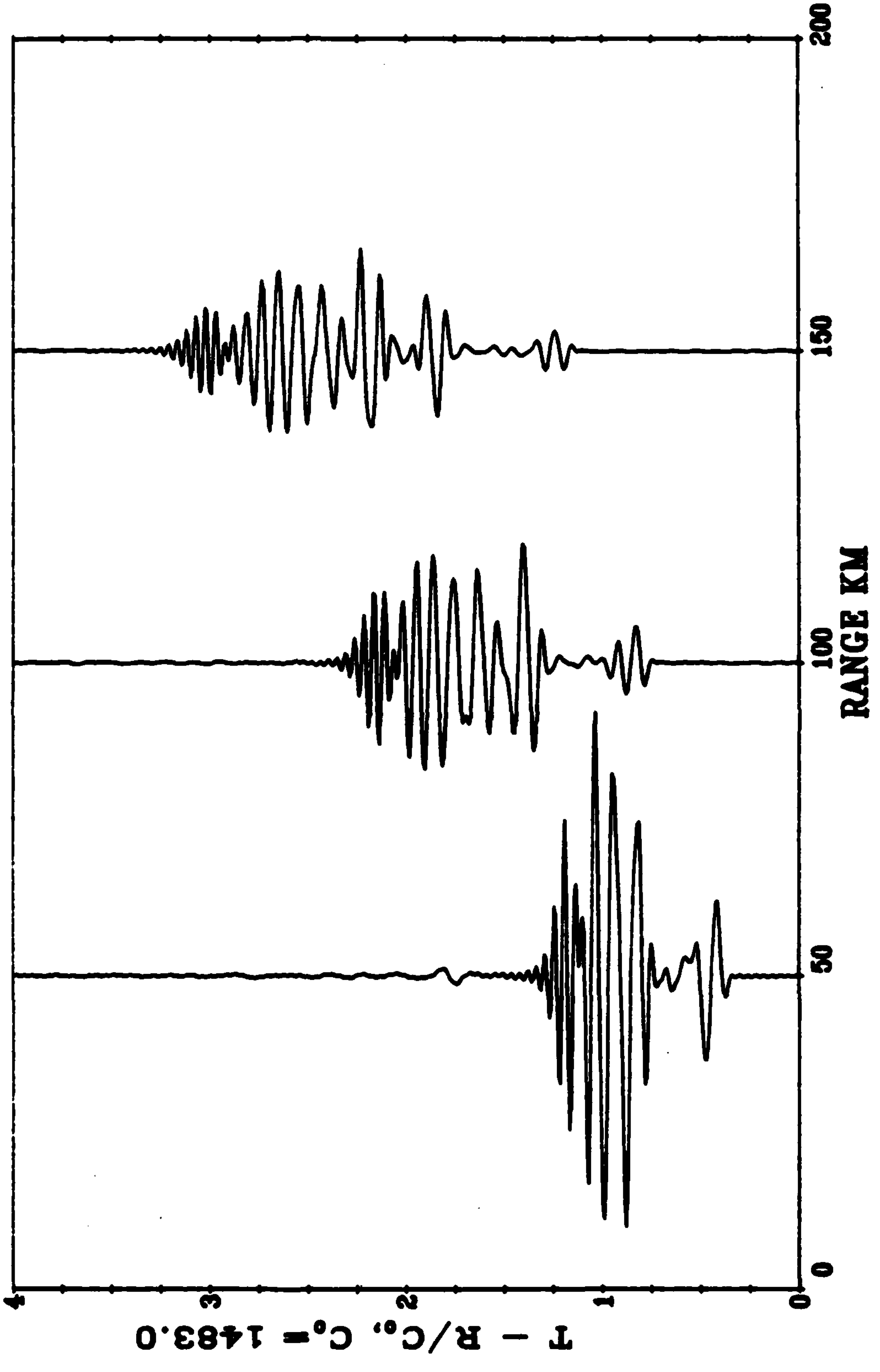

T − R/C₀, C₀= 1499.0
0
1
2
45
50
55
60
65
70
75
80
85
90
95
100
105
110
115
120
125
130
135
140
145
150
155
RANGE KM

```
                    TEST CASE DATA

129   0   1128 12   0   3   0   1   1  25
   50.0      050.0          100.        4200.00    1483.0      1.0         1.0         100.0
 1434.0      1456.          900.0
 2.441407     512         0.008          8.0       10000.0     24.41407
 000.0       1434.0                    .000005                1.0
 050.0       1434.0                    .000005                1.0
 100.0       1437.0                    .000005                1.0
 150.0       1440.0                    .000005                1.0
 200.0       1443.0                    .000005                1.0
 250.0       1446.4                    .000005                1.0
 300.0       1450.0                    .000005                1.0
 350.0       1453.2                    .000005                1.0
 600.0       1456.0                    .000005                1.0
 800.0       1460.0                    .000005                1.0
 1000.       1463.0                    .000005                1.0
 1280.       1467.0                    .000005                1.0
 1500.       1472.0                    .000100                1.0
 1760.       1476.0                    .000100                1.0
 2000.       1480.0                    .000100                1.0
 2280.       1484.0                    .000100                1.0
 2600.       1489.0                    .001000                1.0
 2840.       1494.0                    .001000                1.0
 3040.       1498.0                    .001000                1.0
 3240.       1502.0                    .010000                1.0
 3520.       1506.0                    .010000                1.0
 3760.       1511.0                    .010000                1.0
 3900.       1515.0                    .010000                1.0
 4000.       1600.0                    .010000                1.0
 4200.       1600.0                    .010000                1.0
```


T − R/C₀, C₀= 1483.0
0
1
2
3
4
0
50
100
150
200
RANGE KM

TEST CASE DATA

```
129  0  2128 12  0  3  0  1  1 25
  50.0     050.0        100.      4200.00    1483.0    1.0        1.0       100.0
 1434.0     1456.        075.0      900.0
 2.441407    512       0.008        8.0      10000.0    24.41407
 000.0     1434.0                 .000005              1.0
 050.0     1434.0                 .000005              1.0
 100.0     1437.0                 .000005              1.0
 150.0     1440.0                 .000005              1.0
 200.0     1443.0                 .000005              1.0
 250.0     1446.4                 .000005              1.0
 300.0     1450.0                 .000005              1.0
 350.0     1453.2                 .000005              1.0
 600.0     1456.0                 .000005              1.0
 800.0     1460.0                 .000005              1.0
 1000.     1463.0                 .000005              1.0
 1280.     1467.0                 .000005              1.0
 1500.     1472.0                 .000100              1.0
 1760.     1476.0                 .000100              1.0
 2000.     1480.0                 .000100              1.0
 2280.     1484.0                 .000100              1.0
 2600.     1489.0                 .001000              1.0
 2840.     1494.0                 .001000              1.0
 3040.     1498.0                 .001000              1.0
 3240.     1502.0                 .010000              1.0
 3520.     1506.0                 .010000              1.0
 3760.     1511.0                 .010000              1.0
 3900.     1515.0                 .010000              1.0
 4000.     1600.0                 .010000              1.0
 4200.     1600.0                 .010000              1.0
 000.0     1434.0                 .000005              1.0
 050.0     1437.0                 .000005              1.0
 100.0     1443.0                 .000005              1.0
 150.0     1449.0                 .000005              1.0
 200.0     1450.0                 .000005              1.0
 250.0     1451.0                 .000005              1.0
 300.0     1452.0                 .000005              1.0
 350.0     1453.2                 .000005              1.0
 600.0     1456.0                 .000005              1.0
 800.0     1460.0                 .000005              1.0
 1000.     1463.0                 .000005              1.0
 1280.     1467.0                 .000005              1.0
 1500.     1472.0                 .000100              1.0
 1760.     1476.0                 .000100              1.0
 2000.     1480.0                 .000100              1.0
 2280.     1484.0                 .000100              1.0
 2600.     1489.0                 .001000              1.0
 2840.     1494.0                 .001000              1.0
 3040.     1498.0                 .001000              1.0
 3240.     1502.0                 .010000              1.0
 3520.     1506.0                 .010000              1.0
 3760.     1511.0                 .010000              1.0
 3900.     1515.0                 .010000              1.0
 4000.     1600.0                 .010000              1.0
 4200.     1600.0                 .010000              1.0
```


RANGE KM
0
50
100
150
200
T - R/C$_0$, C$_0$= 1483.0
0
1
2
3
4

Unclassified

SECURITY CLASSIFICATION OF THIS PAGE (*When Data Entered*)

REPORT DOCUMENTATION PAGE

READ INSTRUCTIONS
BEFORE COMPLETING FORM

1. REPORT NUMBER
N00014-80-C-0021

2. GOVT ACCESSION NO.
A147 492

3. RECIPIENT'S CATALOG NUMBER

4. TITLE (*and Subtitle*)
ARCTIC MARINE ACOUSTICS - Final Report Under Contract N00014-80-C-0021

5. TYPE OF REPORT & PERIOD COVERED
FINAL 10/1/79 - 9/30/83

6. PERFORMING ORG. REPORT NUMBER
S-20081

7. AUTHOR(s)
Henry W. Kutschale

8. CONTRACT OR GRANT NUMBER(s)
N00014-80-C-0021

9. PERFORMING ORGANIZATION NAME AND ADDRESS
Lamont-Doherty Geological Observatory
of Columbia University
Palisades, New York 10964

10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS

11. CONTROLLING OFFICE NAME AND ADDRESS
Office of Naval Research
Arctic Programs, Code 425AR
Arlington, Virginia 22217

12. REPORT DATE
October 1984

13. NUMBER OF PAGES
231

14. MONITORING AGENCY NAME & ADDRESS(*if different from Controlling Office*)

15. SECURITY CLASS. (*of this report*)
Unclassified

15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (*of this Report*)

Approved for public release, distribution unlimited.

17. DISTRIBUTION STATEMENT (*of the abstract entered in Block 20, if different from Report*)

18. SUPPLEMENTARY NOTES

19. KEY WORDS (*Continue on reverse side if necessary and identify by block number*)

Arctic Ice, Long-Range Propagation, SOFAR, Bottom Interaction, Frequency Dispersion, Pulse Propagation, Transmission Loss, Pulse Compression, Layered Elastic Media, Normal Modes, Branch-Line Integral, Fast Field Program, Parabolic Equation, Computer Codes, Users Manual

20. ABSTRACT (*Continue on reverse side if necessary and identify by block number*)

Wave-theoretical computer codes have been developed to model pulse propagation in the central Arctic Ocean. Pulse shapes as a function of range and depth are computed from the Pulse Fast Field Program (PFFP) and the pulse parabolic equation (PPE) code. Group- and phase-velocity dispersion and eigenfunctions are computed from the PFFP or from a corresponding normal-mode program. Good agreement has been obtained between measured and computed SOFAR signals. The effect of ice roughness on Arctic SOFAR propagation is illus-

Unclassified

SECURITY CLASSIFICATION OF THIS PAGE (*When Data Entered*)

20.

trated from field data and the PFFP. Hydroacoustic signals from underwater explosions that have propagated over the Arctic abyssal plains commonly display marked frequency dispersion in pulses that are bottom-interacting and that arrive after the SOFAR signal. In the infrasonic band of 2 to 20 Hz, the temporal dispersion for each pulse that has interacted with the flat bottom of the plain can be nearly as strong as that observed in the SOFAR signal for the first mode. However, the bottom-interacting pulses correspond to a coherent summation of many higher-order normal modes in a channel bounded above by the ocean surface and below by the zone of increasing velocity in the upper 400 m of the bottom sediment. Using normal-mode theory and the Multiple Scattering Pulse Fast Field Program (MSPFFP), we have analyzed the dispersion and pulse shapes and have derived the acoustic properties of the bottom in the Pole, Barents, and Mendeleyev Abyssal Plains. The principal properties of the bottom controlling the propagation are compressional velocity, density, and attentuation. In contrast, the ice layer has a negligible effect on the dispersion of the observed waves. The effect on pulse compression of this frequency dispersion of the bottom-interacting signals was simulated numerically, using predistorted waveforms matched to the dispersion of the SOFAR channel at specified ranges. We include a users manual (with test computations) for our computer codes. We hope that this manual will be useful to the many users of our codes.